10-, 12-, 14-Bit, 1200 MSPS DACS

AD9734/AD9735/AD9736

FEATURES

Pin-compatible family
Excellent dynamic performance
- **AD9736: SFDR = 82 dBc at f_{OUT} = 30 MHz**
- **AD9736: SFDR = 69 dBc at f_{OUT} = 130 MHz**
- **AD9736: IMD = 87 dBc at f_{OUT} = 30 MHz**
- **AD9736: IMD = 82 dBc at f_{OUT} = 130 MHz**

LVDS data interface with on-chip 100 Ω terminations
Built-in self test
- **LVDS sampling integrity**
- **LVDS-to-DAC data transfer integrity**

Low power: 380 mW (I_{FS} = 20 mA; f_{OUT} = 330 MHz)
1.8/3.3 V dual-supply operation
Adjustable analog output
- **8.66 mA to 31.66 mA (RL = 25 Ω to 50 Ω)**

On-chip 1.2 V reference
160-lead chip scale ball grid array (CSP_BGA) package

APPLICATIONS

Broadband communications systems
- **Cellular infrastructure (digital predistortion)**
- **Point-to-point wireless**
- **CMTS/VOD**

Instrumentation, automatic test equipment
Radar, avionics

PRODUCT DESCRIPTION

The AD9736, AD9735, and AD9734 are high performance, high frequency DACs that provide sample rates of up to 1200 MSPS, permitting multicarrier generation up to their Nyquist frequency. The AD9736 is the 14-bit member of the family, while the AD9735 and the AD9734 are the 12-bit and 10-bit members, respectively. They include a serial peripheral interface (SPI) port that provides for programming of many internal parameters and also enables readback of status registers. A reduced-specification LVDS interface is utilized to achieve the high sample rate. The output current can be programmed over a range of 8.66 mA to 31.66 mA. The AD973x family is manufactured on a 0.18 μm CMOS process and operates from 1.8 V and 3.3 V supplies for a total power consumption of 380 mW in bypass mode. It is supplied in a 160-lead chip scale ball grid array for reduced package parasitics.

FUNCTIONAL BLOCK DIAGRAM

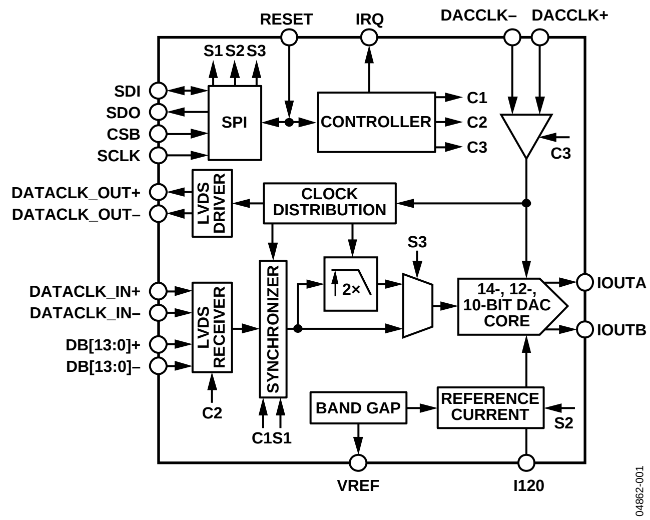


Figure 1.

PRODUCT HIGHLIGHTS

1. Low noise and intermodulation distortion (IMD) features enable high quality synthesis of wideband signals at intermediate frequencies up to 600 MHz.
2. Double data rate (DDR) LVDS data receivers support the maximum conversion rate of 1200 MSPS.
3. Direct pin programmability of basic functions or SPI port access for complete control of all AD973x family functions.
4. Manufactured on a CMOS process, the AD973x family uses a proprietary switching technique that enhances dynamic performance.
5. The current output(s) of the AD9736 family are easily configured for single-ended or differential circuit topologies.

Rev. 0

One Technology Way, P.O. Box 9106, Norwood, MA 02062-9106, U.S.A.
Tel: 781.329.4700 **www.analog.com**
Fax: 781.461.3113

TABLE OF CONTENTS

REVISION HISTORY

4/05—Revision 0: Initial Version

DC SPECIFICATIONS

AVDD33 = DVDD33 = 3.3 V, CVDD18 = DVDD18 = 1.8 V, maximum sample rate, I_{FS} = 20 mA, 1× mode, 25 Ω 1% balanced load, unless otherwise noted.

Table 1.

	AD9736			AD9735			AD9734			
Parameter	**Min**	**Typ**	**Max**	**Min**	**Typ**	**Max**	**Min**	**Typ**	**Max**	**Unit**
RESOLUTION		14			12			10		Bits
ACCURACY										
Integral Nonlinearity (INL)	−5.6	±1.0	+5.6	−1.5	±0.50	+1.5	−0.5	±0.12	+0.5	LSB
Differential Nonlinearity (DNL)	−2.1	±0.6	+2.1	−0.5	±0.25	+0.5	−0.1	±0.06	+0.1	LSB
ANALOG OUTPUTS										
Offset Error	−0.01	±0.005	+0.01	−0.01	±0.005	+0.01	−0.01	±0.005	+0.01	% FSR
Gain Error (With Internal Reference)		±1.0			±1.0			±1.0		% FSR
Gain Error (Without Internal Reference)		±1.0			±1.0			±1.0		% FSR
Full-Scale Output Current	8.66	20.2	31.66	8.66	20.2	31.66	8.66	20.2	31.66	mA
Output Compliance Range	−1.0		+1.0	−1.0		1.0	−1.0		+1.0	V
Output Resistance		10			10			10		MΩ
Output Capacitance		1			1			1		pF
TEMPERATURE DRIFT										
Offset		0			0			0		ppm/°C
Gain		80			80			80		ppm/°C
Reference Voltage[1]		40			40			40		ppm/°C
REFERENCE										
Internal Reference Voltage[1]	1.14	1.2	1.26	1.14	1.2	1.26	1.14	1.2	1.26	V
Output Resistance[2]		5			5			5		kΩ
ANALOG SUPPLY VOLTAGES										
AVDD33	3.13	3.3	3.47	3.13	3.3	3.47	3.13	3.3	3.47	V
CVDD18	1.70	1.8	1.90	1.70	1.8	1.90	1.70	1.8	1.90	V
DIGITAL SUPPLY VOLTAGES										
DVDD33	3.13	3.3	3.47	3.13	3.3	3.47	3.13	3.3	3.47	V
DVDD18	1.70	1.8	1.90	1.70	1.8	1.90	1.70	1.8	1.90	V
SUPPLY CURRENTS										
1× Mode, 1.2 GSPS										
I_{AVDD33}		25			25			25		mA
I_{CVDD18}		47			47			47		mA
I_{DVDD33}		10			10			10		mA
I_{DVDD18}		122			122			122		mA
FIR Bypass (1×) Mode		380			380			380		mW
2× Mode, 1.2 GSPS										
I_{AVDD33}		25			25			25		mA
I_{CVDD18}		47			47			47		mA
I_{DVDD33}		10			10			10		mA
I_{DVDD18}		234			234			234		mA
FIR 2× Interpolation Filter Enabled		550			550			550		mW

	AD9736			AD9735			AD9734			
Parameter	**Min**	**Typ**	**Max**	**Min**	**Typ**	**Max**	**Min**	**Typ**	**Max**	**Unit**
Static, No Clock										
I_{AVDD33}		25			25			25		mA
I_{CVDD18}		8			8			8		mA
I_{DVDD33}		10			10			10		mA
I_{DVDD18}		2			2			2		mA
FIR Bypass (1×) Mode		133			133			133		mW
Sleep Mode, No Clock										
I_{AVDD33}		2.5	3.15		2.5	3.15		2.5	3.15	mA
FIR Bypass (1×) Mode		59	65		59	65		59	65	mW
Power-Down Mode										
I_{AVDD33}		0.01	0.13		0.01	0.13		0.01	0.13	mA
I_{CVDD18}		0.02	0.12		0.02	0.12		0.02	0.12	mA
I_{DVDD33}		0.01	0.12		0.01	0.12		0.01	0.12	mA
I_{DVDD18}		0.01	0.11		0.01	0.11		0.01	0.11	mA
FIR Bypass (1×) Mode		0.12	1.24		0.12	1.24		0.12	1.24	mW

[1] Default band gap adjustment (Reg0E<2:0> = 0h).
[2] Use an external amplifier to drive any external load.

DIGITAL SPECIFICATIONS

AVDD33 = DVDD33 = 3.3 V, CVDD18 = DVDD18 = 1.8 V, maximum sample rate, I_{FS} = 20 mA, 1× mode, 25 Ω 1% balanced load, unless otherwise noted.

LVDS drivers and receivers are compliant to the IEEE-1596 reduced range link, unless otherwise noted.

Table 2.

Parameter	Min	Typ	Max	Unit
LVDS DATA INPUTS (DB[13:0]+, DB[13:0]−) DB+ = V_{ia}, DB- = V_{ib}				
Input Voltage Range, V_{ia} or V_{ib}	825		1575	mV
Input Differential Threshold, V_{idth}	−100		+100	mV
Input Differential Hysteresis, $V_{idthh} - V_{idthl}$		20		mV
Receiver Differential Input Impedance, R_{in}	80		120	Ω
LVDS Input Rate	1200			MSPS
LVDS Minimum Data Valid Period (t_{MDE})			344	ps
LVDS CLOCK INPUT (DATACLK_IN+, DATACLK_IN−) DATACLK_IN+ = V_{ia}, DATACLK_IN− = V_{ib}				
Input Voltage Range, V_{ia} or V_{ib}	825		1575	mV
Input Differential Threshold[1], V_{idth}	−100		+100	mV
Input Differential Hysteresis, V_{idthh} - V_{idthl}		20		mV
Receiver Differential Input Impedance, R_{in}	80		120	Ω
Maximum Clock Rate	600			MHz
LVDS CLOCK OUTPUT (DATACLK_OUT+, DATACLK_ OUT−) DATACLK_OUT+ = V_{oa}, DATACLK_OUT− = V_{ob} 100 Ω Termination				
Output Voltage High, V_{oa} or V_{ob}			1375	mV
Output Voltage Low, V_{oa} or V_{ob}	1025			mV
Output Differential Voltage, $\|V_{od}\|$	150	200	250	mV
Output Offset Voltage, V_{os}	1150		1250	mV
Output Impedance, Single-Ended, R_o	80	100	120	Ω
Ro Mismatch Between A and B, ΔR_o			10	%
Change in |Vod| Between 0 and 1, $\|\Delta V_{od}\|$			25	mV
Change in Vos Between 0 and 1, ΔV_{os}			25	mV
Output Current—Driver Shorted to Ground, I_{sa}, I_{sb}			20	mA
Output Current—Drivers Shorted Together, I_{sab}			4	mA
Power-Off Output Leakage, $\|I_{xa}\|$, $\|I_{xb}\|$			10	mA
Maximum Clock Rate	600			MHz
DAC CLOCK INPUT (CLK+, CLK−)				
Input Voltage Range, CLK− or CLK+	0		800	
Differential Peak-to-Peak Voltage	400	800	1600	mV
Common-Mode Voltage	300	400	500	mV
Maximum Clock Rate	1200			MHz
SERIAL PERIPHERAL INTERFACE				
Maximum Clock Rate (f_{SCLK}, $1/t_{SCLK}$)			20	MHz
Minimum Pulse Width High, t_{PWH}	20			ns
Minimum Pulse Width Low, t_{PWL}	20			ns
Minimum SDIO and CSB to SCLK Setup, t_{DS}		10		ns
Minimum SCLK to SDIO Hold, t_{DH}		5		ns
Maximum SCLK to Valid SDIO and SDO, t_{DV}		20		ns
Minimum SCLK to Invalid SDIO and SDO, t_{DNV}		5		ns

Parameter	Min	Typ	Max	Unit
INPUTS (SDI, SDIO, SCLK, CSB)				
Voltage in High, V_{ih}	2.0	3.3		V
Voltage in Low, V_{il}		0	0.8	V
Current in High, I_{ih}	−10		+10	µA
Current in Low, I_{il}	−10		+10	µA
Input Capacitance				pF
SDIO OUTPUT				
Voltage out High, V_{oh}	2.4		3.6	V
Voltage out Low, V_{ol}	0		0.4	V
Current out High, I_{oh}		4		mA
Current out Low, I_{ol}		4		mA

[1] Refer to the Input Data Timing section for recommended LVDS differential drive levels.

AC SPECIFICATIONS

AVDD33 = DVDD33 = 3.3 V, CVDD18 = DVDD18 = 1.8 V, maximum sample rate, I_{FS} = 20 mA, 1× mode, 25 Ω 1% balanced load, unless otherwise noted.

Table 3.

	AD9736			AD9735			AD9734			
Parameter	**Min**	**Typ**	**Max**	**Min**	**Typ**	**Max**	**Min**	**Typ**	**Max**	**Unit**
DYNAMIC PERFORMANCE										
Maximum Update Rate	1200			1200			1200			MSPS
SPURIOUS-FREE DYNAMIC RANGE (SFDR)										
f_{DAC} = 800 MSPS										
f_{OUT} = 20 MHz		75			75			75		dBc
f_{DAC} = 1200 MSPS										
f_{OUT} = 50 MHz		80			76			76		dBc
f_{OUT} = 100 MHz		77			74			71		dBc
f_{OUT} = 316 MHz		63			63			60		dBc
f_{OUT} = 550 MHz		55			54			53		dBc
TWO-TONE INTERMODULATION DISTORTION (IMD)										
f_{DAC} = 1200 MSPS										
f_{OUT2} = f_{OUT} + 1.25 MHz										
f_{OUT} = 40 MHz		88			84			83		dBc
f_{OUT} = 50 MHz		85			84			83		dBc
f_{OUT} = 100 MHz		84			81			79		dBc
f_{OUT} = 315 MHz		70.5			67			66		dBc
f_{OUT} = 550 MHz		65			60			60		dBc
NOISE SPECTRAL DENSITY (NSD)										
Single Tone										
f_{DAC} = 1200 MSPS										
f_{OUT} = 50 MHz		−165			−162			−154		dBm/Hz
f_{OUT} = 100 MHz		−164			−161			−154		dBm/Hz
f_{OUT} = 241MHz	−158.5	−160.5			−159.5			−155		dBm/Hz
f_{OUT} = 316 MHz		−158			−157			−152		dBm/Hz
f_{OUT} = 550 MHz		−155			−155			−149		dBm/Hz
Eight-Tone										
f_{DAC} = 1200 MSPS, 500 kHz Tone Spacing										
f_{OUT} = 50 MHz		−166.5			−163			−154		dBm/Hz
f_{OUT} = 100 MHz		−166			−163			−152		dBm/Hz
f_{OUT} = 241MHz	−163.3	−165			−161.5			−150.5		dBm/Hz
f_{OUT} = 316 MHz		−164			−162			−151		dBm/Hz
f_{OUT} = 550 MHz		−162			−160			−150		dBm/Hz

ABSOLUTE MAXIMUM RATINGS

Table 4.

Parameter	With Respect to	Min	Max
AVDD33	AVSS	−0.3 V	+3.6 V
DVDD33	DVSS	−0.3 V	+3.6 V
DVDD18	DVSS	−0.3 V	+1.98 V
CVDD18	CVSS	−0.3 V	+1.98 V
AVSS	DVSS	−0.3 V	+0.3 V
AVSS	CVSS	−0.3 V	+0.3 V
DVSS	CVSS	−0.3 V	+0.3 V
CLK+, CLK−	CVSS	−0.3 V	CVDD18 + 0.18 V
PIN_MODE	DVSS	−0.3 V	DVDD33 + 0.3 V
DATACLK_IN, DATACLK_OUT	DVSS	−0.3 V	DVDD33 + 0.3 V
LVDS Data Inputs	DVSS	−0.3 V	DVDD33 + 0.3 V
IOUTA, IOUTB	AVSS	−1.0 V	AVDD33 + 0.3 V
I120, VREF, IPTAT	AVSS	−0.3 V	AVDD33 + 0.3 V
IRQ, CSB, SCLK, SDO, SDIO, RESET	DVSS	−0.3 V	DVDD33 + 0.3 V
Junction Temp.			150°C
Storage Temp.		−65°C	+150°C

Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only; functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum ratings for extended periods may effect device reliability.

ESD CAUTION

ESD (electrostatic discharge) sensitive device. Electrostatic charges as high as 4000 V readily accumulate on the human body and test equipment and can discharge without detection. Although this product features proprietary ESD protection circuitry, permanent damage may occur on devices subjected to high energy electrostatic discharges. Therefore, proper ESD precautions are recommended to avoid performance degradation or loss of functionality.



Note that this device in its current form does not meet Analog Devices' standard requirements for ESD as measured against the charged device model (CDM). As such, special care should be used when handling this product, especially in a manufacturing environment. Analog Devices will provide a more ESD-hardy product in the near future at which time this warning will be removed from this datasheet.

PIN CONFIGURATION AND FUNCTION DESCRIPTIONS

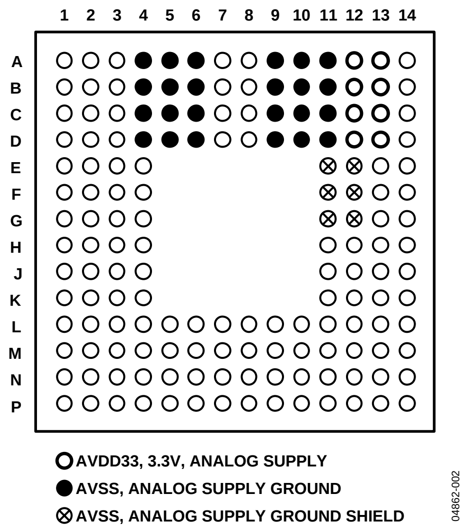


Figure 2. AD9736 Analog Supply Pins (Top View)

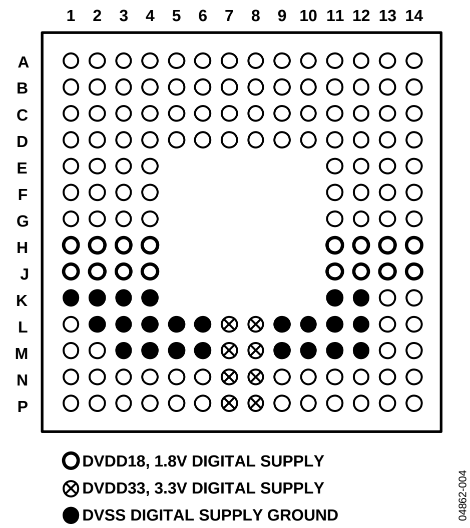


Figure 4. AD9736 Digital Supply Pins (Top View)

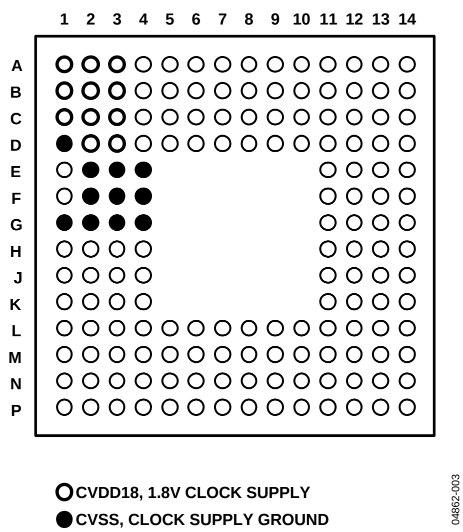


Figure 3. AD9736 Clock Supply Pins (Top View)

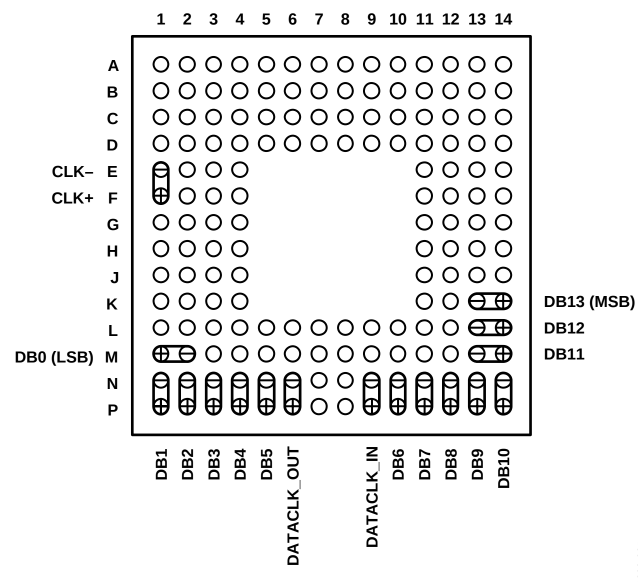


Figure 5. AD9736 Digital LVDS Inputs, Clock I/O (Top View)

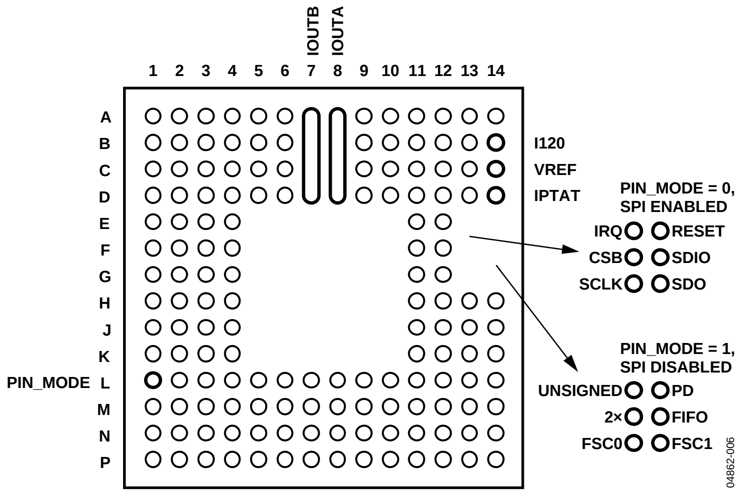


Figure 6. AD9736 Analog I/O and SPI Control Pins (Top View)

Table 5. Pin Function Descriptions

Pin No.	Mnemonic	Description
A1, A2, A3, B1, B2, B3, C1, C2, C3, D2, D3	CVDD18	1.8 V Clock Supply.
A4, A5, A6, A9, A10, A11, B4, B5, B6, B9, B10, B11, C4, C5, C6, C9, C10, C11, D4, D5, D6, D9, D10, D11	AVSS	Analog Supply Ground.
A7, B7, C7, D7	IOUTB	DAC Negative Output; 10 mA to 30 mA full-scale output current.
A8, B8, C8, D8	IOUTA	DAC Positive Output; 10 mA to 30 mA full-scale output current.
A12, A13, B12, B13, C12, C13, D12, D13	AVDD33	3.3 V Analog Supply.
A14	DNC	Do Not Connect.
B14	I120	Nominal 1.2 V Reference; tie to analog ground via 10 kΩ resistor to generate a 120 μA reference current.
C14	VREF	Band Gap Voltage Reference I/O; tie to analog ground via 1 nF capacitor, output impedance approximately 5 kΩ.
D1, E2, E3, E4, F2, F3, F4, G1, G2, G3, G4	CVSS	Clock Supply Ground.
D14	IPTAT	Factory Test Pin; output current proportional to absolute temperature, approximately 10 μA at 25°C with approximately 20 nA/°C slope.
E1, F1	DACCLK−/DACCLK+	Negative/Positive DAC Clock Input (DACCLK).
E11, E12, F11, F12, G11, G12	AVSS	Analog Supply Ground Shield; tie to AVSS at the DAC.
E13	IRQ/UNSIGNED	If PIN_MODE = 0, IRQ: Active low open-drain interrupt request output, pull up to DVDD33 with 10 kΩ resistor. If PIN_MODE = 1, UNSIGNED: Digital input pin where 0 = twos complement input data format, 1 = unsigned.
E14	RESET/PD	If PIN_MODE = 0, RESET: 1 resets the AD9736. If PIN_MODE = 1, PD: 1 puts the AD9736 in the power-down state.
F13	CSB/2×	See Serial Peripheral Interface and Pin Mode Operation sections for pin description.
F14	SDIO/FIFO	See the Pin Mode Operation section for pin description.
G13	SCLK/FSC0	See the Pin Mode Operation section for pin description.
G14	SDO/FSC1	See the Pin Mode Operation section for pin description.
H1, H2, H3, H4, H11, H12, H13, H14, J1, J2, J3, J4, J11, J12, J13, J14	DVDD18	1.8 V Digital Supply.
K1, K2, K3, K4, K11, K12, L2, L3, L4, L5, L6, L9, L10, L11, L12, M3, M4, M5, M6, M9, M10, M11, M12	DVSS	Digital Supply Ground.

Pin No.	Mnemonic	Description
K13, K14	DB<13>−/DB<13>+	Negative/Positive Data Input Bit 13 (MSB); reduced swing LVDS.
L1	PIN_MODE	0 = SPI Mode; SPI enabled. 1 = PIN Mode; SPI disabled, direct pin control.
L7, L8, M7, M8, N7, N8, P7, P8	DVDD33	3.3 V Digital Supply.
L13, L14	DB<12>−/DB<12>+	Negative/Positive Data Input Bit 12; reduced swing LVDS.
M2, M1	DB<0>−/DB<0>+	Negative/Positive Data Input Bit 0 (LSB); reduced swing LVDS.
M13, M14	DB<11>−/DB<11>+	Negative/Positive Data Input Bit 11; reduced swing LVDS.
N1, P1	DB<1>−/DB<1>+	Negative/Positive Data Input Bit 1; reduced swing LVDS.
N2, P2	DB<2>−/DB<2>+	Negative/Positive Data Input Bit 2; reduced swing LVDS.
N3, P3	DB<3>−/DB<3>+	Negative/Positive Data Input Bit 3; reduced swing LVDS.
N4, P4	DB<4>−/DB<4>+	Negative/Positive Data Input Bit 4; reduced swing LVDS..
N5, P5	DB<5>−/DB<5>+	Negative/Positive Data Input Bit 5; reduced swing LVDS.
N6, P6	DATACLK_OUT−/ DATACLK_OUT+	Negative/Positive Data Output Clock; reduced swing LVDS.
N9, P9	DATACLK_IN−/ DATACLK_IN+	Negative/Positive Data Input Clock; reduced swing LVDS
N10, P10	DB<6>−/DB<6>+	Negative/Positive Data Input Bit 6; reduced swing LVDS.
N11, P11	DB<7>−/DB<7>+	Negative/Positive Data Input Bit 7; reduced swing LVDS.
N12, P12	DB<8>−/DB<8>+	Negative/Positive Data Input Bit 8; reduced swing LVDS.
N13, P13	DB<9>−/DB<9>+	Negative/Positive Data Input Bit 9; reduced swing LVDS.
N14, P14	DB<10>−/DB<10>+	Negative/Positive Data Input Bit 10; reduced swing LVDS.

TERMINOLOGY

Linearity Error (Integral Nonlinearity or INL)
The maximum deviation of the actual analog output from the ideal output, determined by a straight line drawn from zero to full scale.

Differential Nonlinearity (DNL)
The measure of the variation in analog value, normalized to full scale, associated with a 1 LSB change in digital input code.

Monotonicity
A DAC is monotonic if the output either increases or remains constant as the digital input increases.

Offset Error
The deviation of the output current from the ideal of zero is called the offset error. For IOUTA, 0 mA output is expected when the inputs are all 0s. For IOUTB, 0 mA output is expected when all inputs are set to 1s.

Gain Error
The difference between the actual and ideal output span. The actual span is determined by the output when all inputs are set to 1s minus the output when all inputs are set to 0s.

Output Compliance Range
The range of allowable voltage at the output of a current output DAC. Operation beyond the maximum compliance limits may cause either output stage saturation or breakdown, resulting in nonlinear performance.

Temperature Drift
Specified as the maximum change from the ambient (25°C) value to the value at either T_{MIN} or T_{MAX}. For offset and gain drift, the drift is reported in ppm of full-scale range (FSR) per °C. For reference drift, the drift is reported in ppm per °C.

Power Supply Rejection
The maximum change in the full-scale output as the supplies are varied from nominal to minimum and maximum specified voltages.

Settling Time
The time required for the output to reach and remain within a specified error band about its final value, measured from the start of the output transition.

Glitch Impulse
Asymmetrical switching times in a DAC give rise to undesired output transients that are quantified by a glitch impulse. It is specified as the net area of the glitch in pV-s.

Spurious-Free Dynamic Range
The difference, in dB, between the rms amplitude of the output signal and the peak spurious signal over the specified bandwidth.

Total Harmonic Distortion (THD)
THD is the ratio of the rms sum of the first six harmonic components to the rms value of the measured input signal. It is expressed as a percentage or in decibels (dB).

Multitone Power Ratio
The spurious-free dynamic range containing multiple carrier tones of equal amplitude. It is measured as the difference between the rms amplitude of a carrier tone to the peak spurious signal in the region of a removed tone.

TYPICAL PERFORMANCE CHARACTERISTICS

AD9736 STATIC LINEARITY, 10 mA FULL SCALE

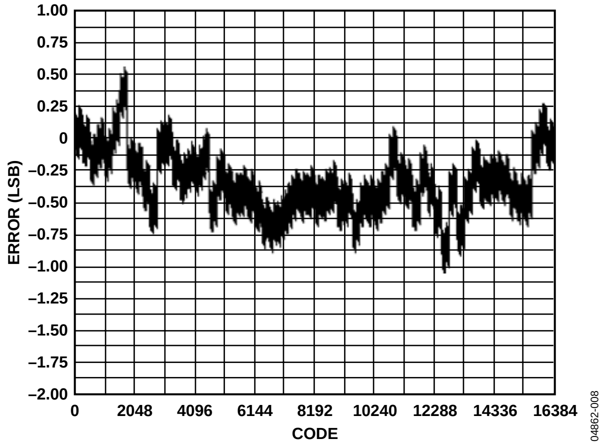


Figure 7. AD9736 INL, −40°C, 10 mA FS

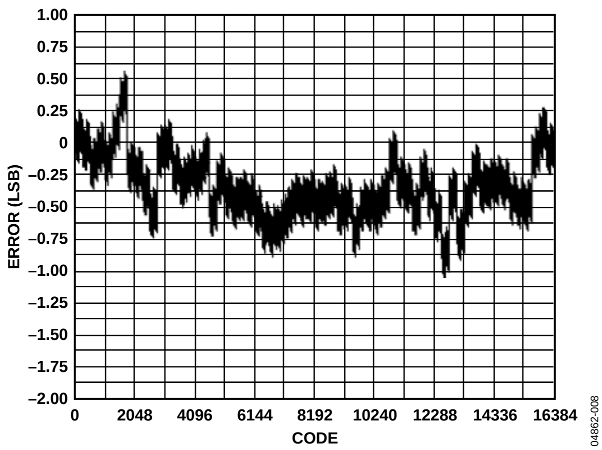


Figure 8. AD9736 INL, 25°C, 10 mA FS

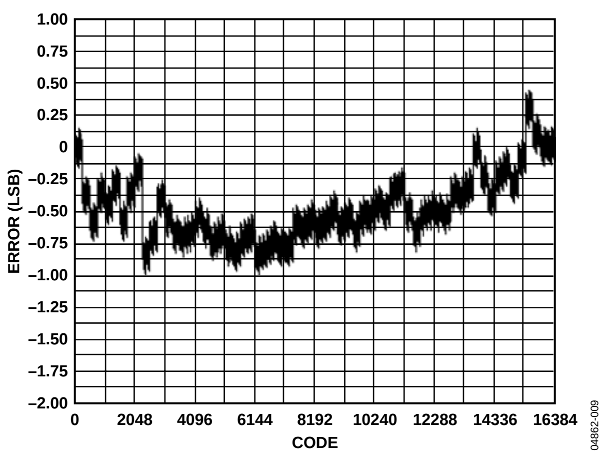


Figure 9. AD9736 INL, 85°C, 10 mA FS

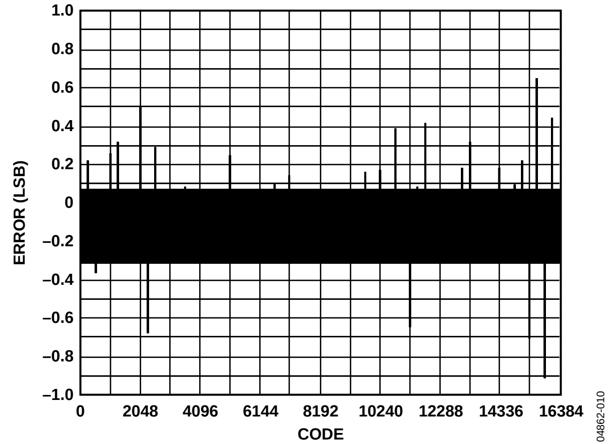


Figure 10. AD9736 DNL, −40°C, 10 mA FS

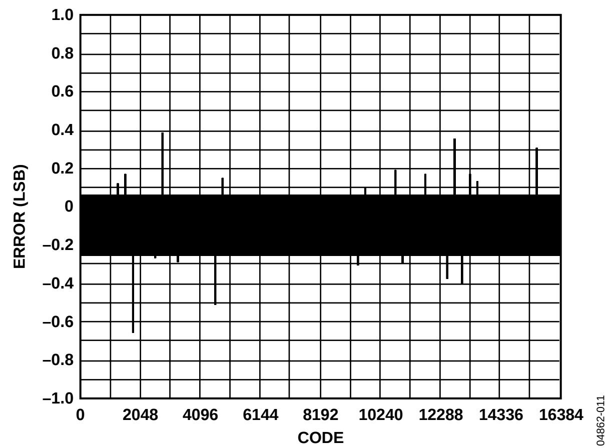


Figure 11. AD976 DNL, 25°C, 10 mA FS

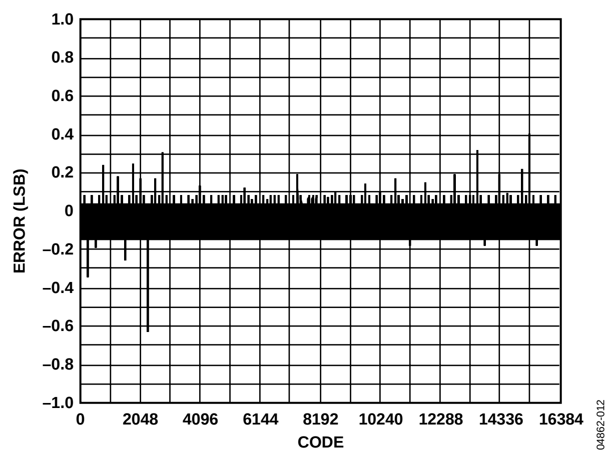


Figure 12. AD9736 DNL, 85°C, 10 mA FS

AD9736 STATIC LINEARITY, 20 mA FULL SCALE

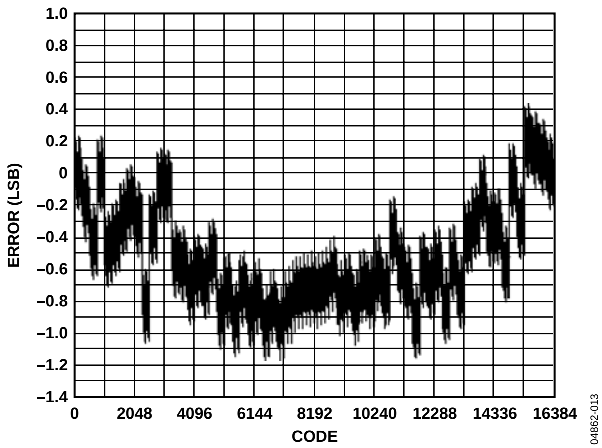


Figure 13. AD9736 INL, −40°C, 20 mA FS

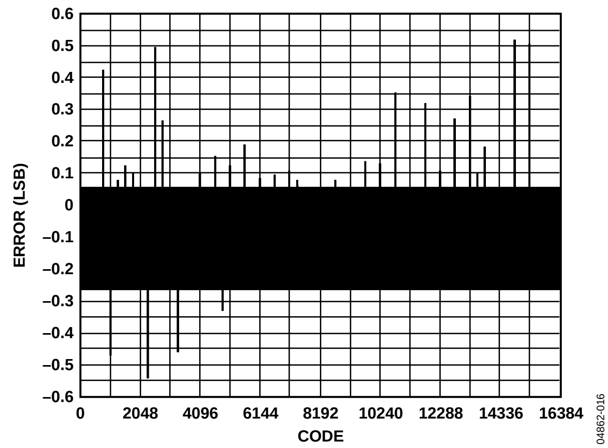


Figure 16. AD9736 DNL, −40°C, 20 mA FS

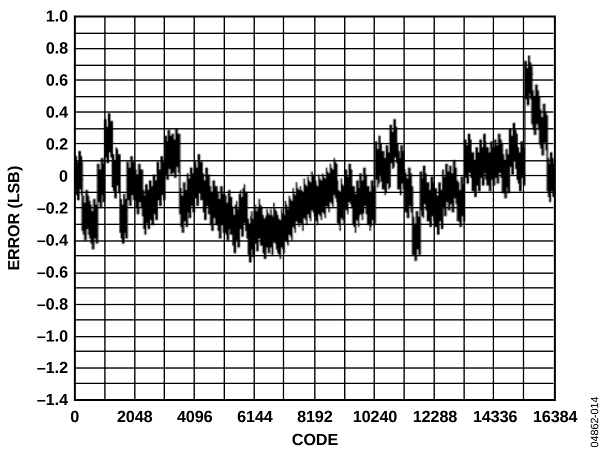


Figure 14. AD9736 INL, 25°C, 20 mA FS

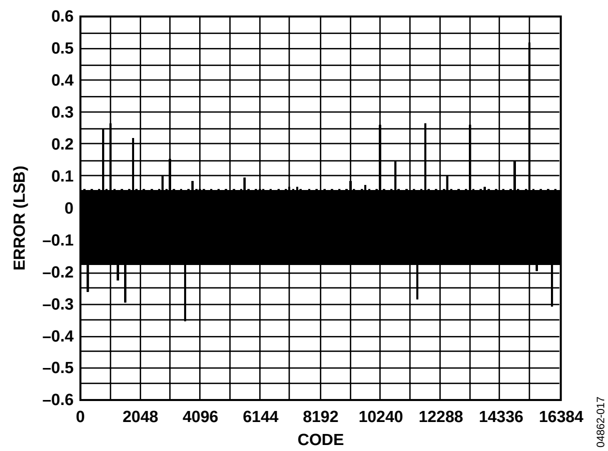


Figure 17. AD9736 DNL, 25°C, 20 mA FS

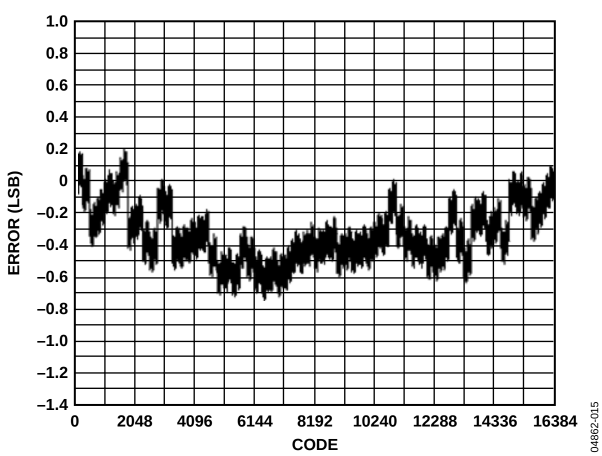


Figure 15. AD9736 INL, 85°C, 20 mA FS

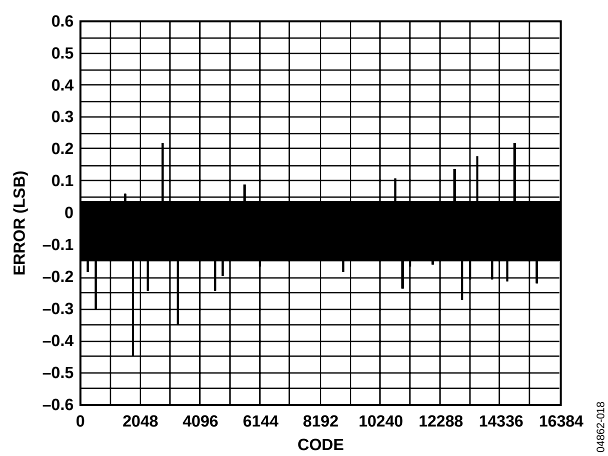


Figure 18. AD9736 DNL, 85°C, 20 mA FS

AD9736 STATIC LINEARITY, 30 mA FULL SCALE

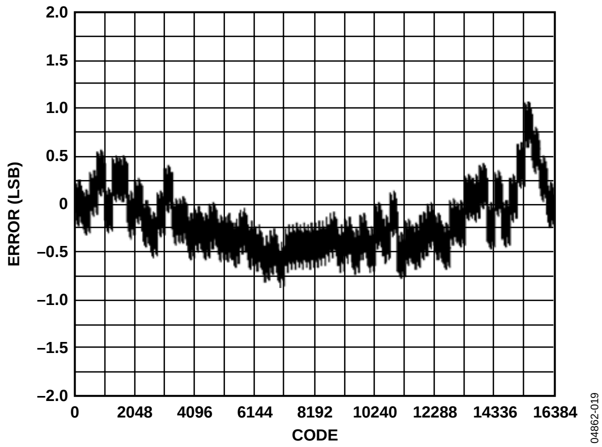


Figure 19. AD9736 INL, −40°C, 30 mA FS

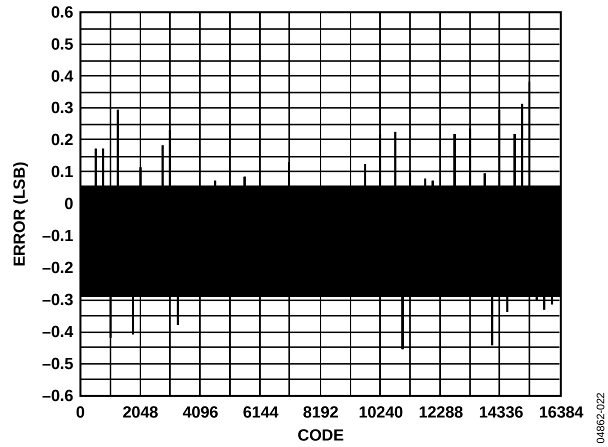


Figure 22. AD9736 DNL, −40°C, 30 mA FS

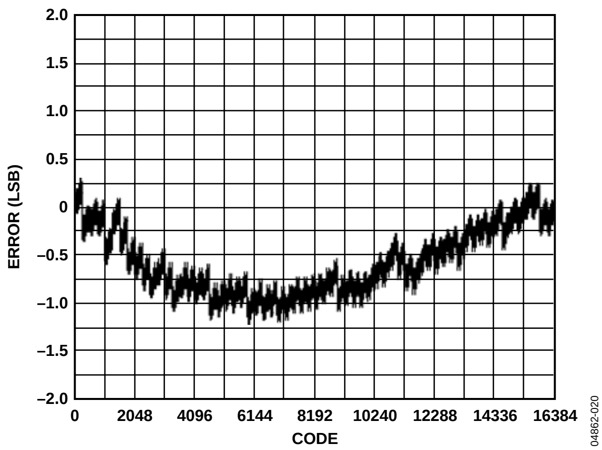


Figure 20. AD9736 INL, 25°C, 30 mA FS

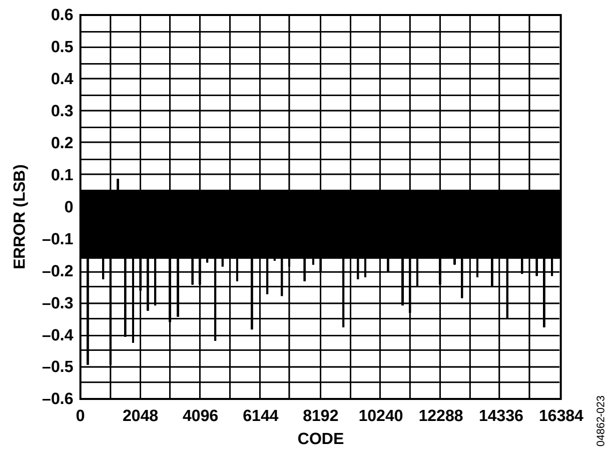


Figure 23. AD9736 DNL, 25°C, 30 mA FS

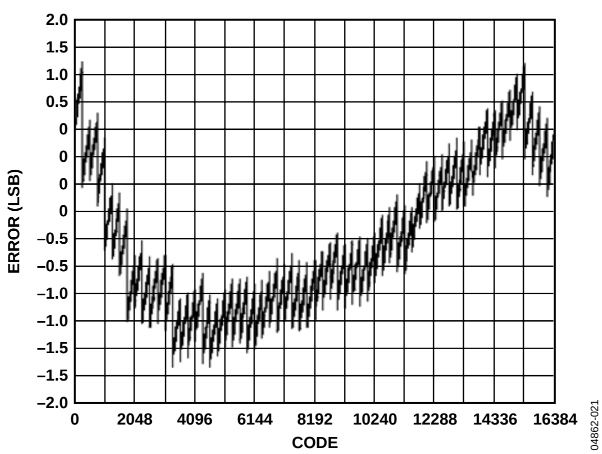


Figure 21. AD9736 INL, 85°C, 30 mA FS

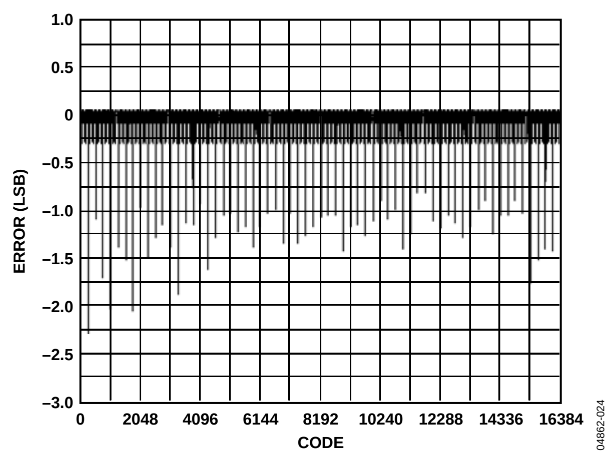


Figure 24. AD9736 DNL, 85°C, 30 mA FS

AD9735 STATIC LINEARITY, 10 mA, 20 mA, 30 mA FULL SCALE

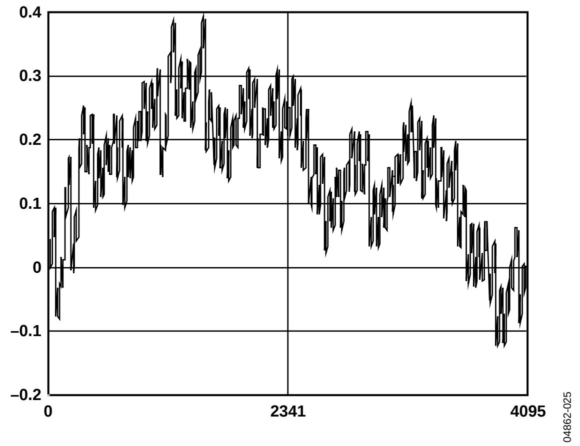


Figure 25. AD9735 INL, 25°C, 10 mA FS

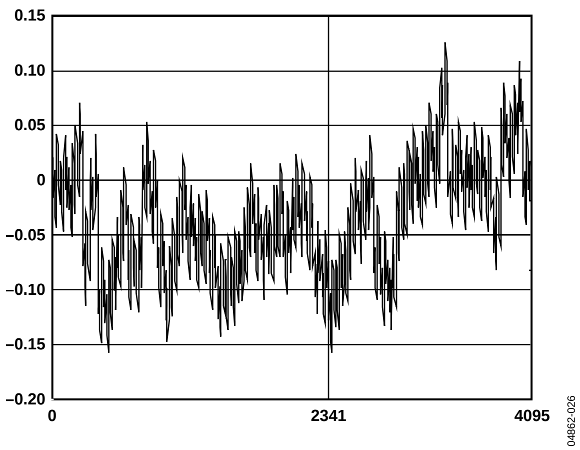


Figure 26. AD9735 INL, 25°C, 20 mA FS

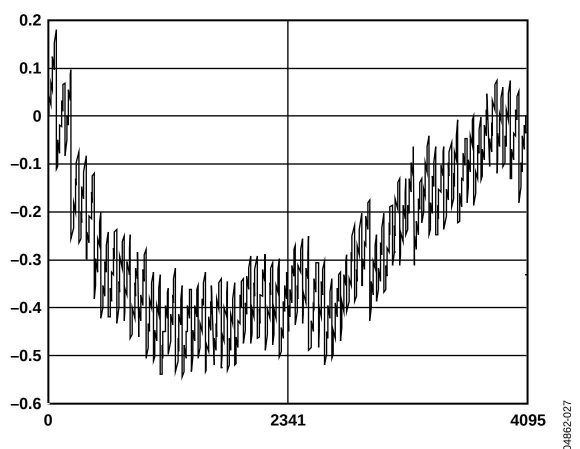


Figure 27. AD9735 INL, 25°C, 30 mA FS

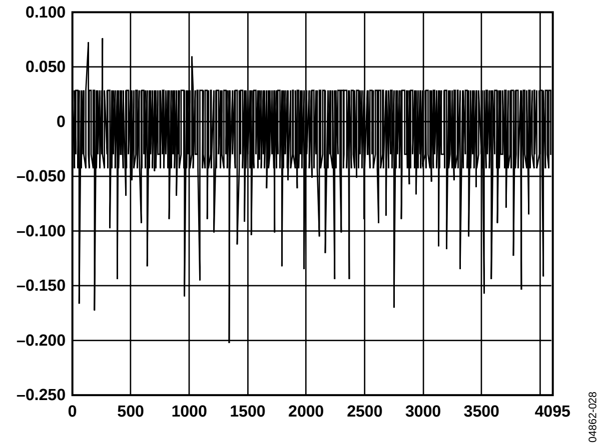


Figure 28. AD9735 DNL, 25°C, 10 mA FS

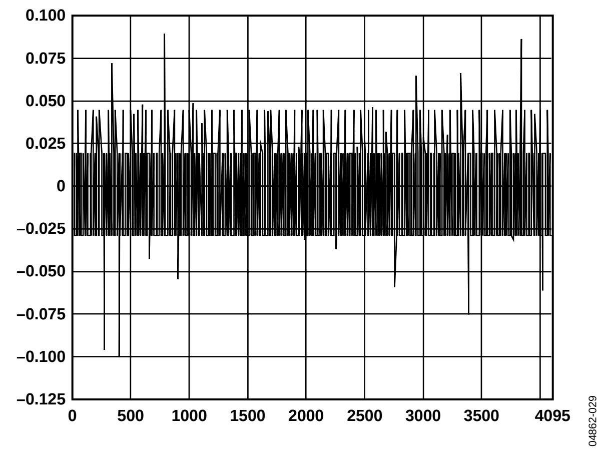


Figure 29. AD9735 DNL, 25°C, 20 mA FS

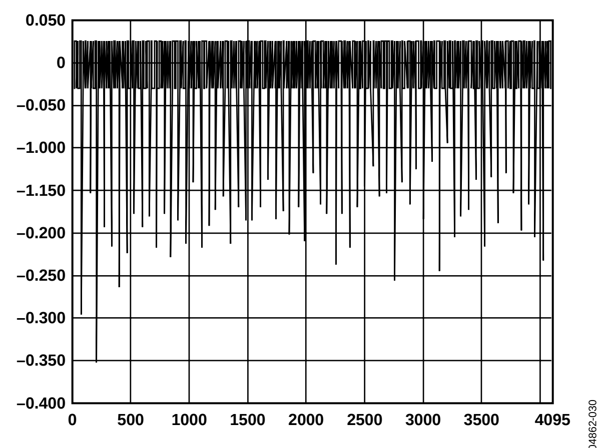


Figure 30. AD9735 DNL, 25°C, 30 mA FS

AD9734 STATIC LINEARITY, 10 mA, 20 mA, 30 mA FULL SCALE

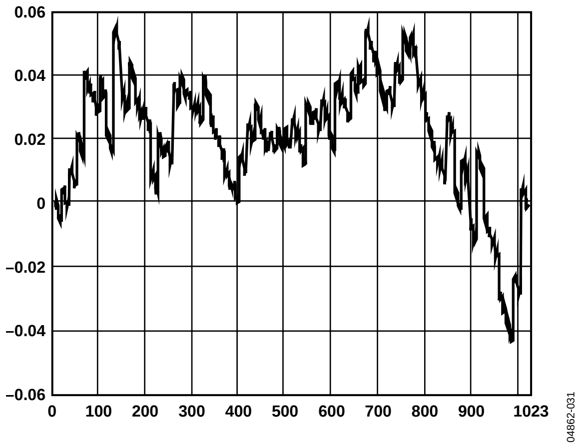


Figure 31. AD9734 INL, 25°C, 10 mA FS

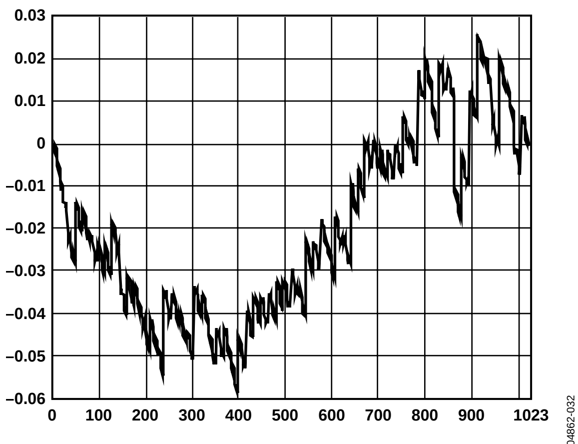


Figure 32. AD9734 INL, 25°C, 20 mA FS

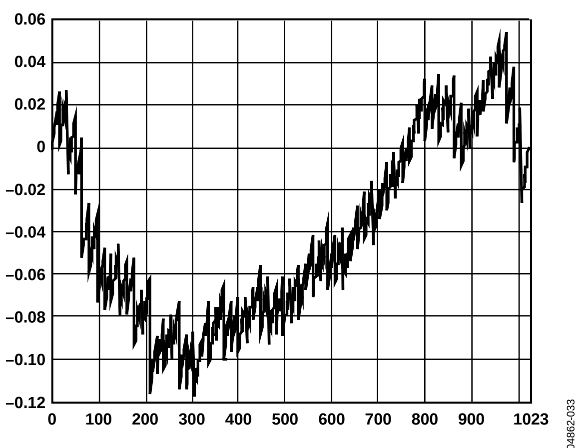


Figure 33. AD9734 INL, 25°C, 30 mA FS

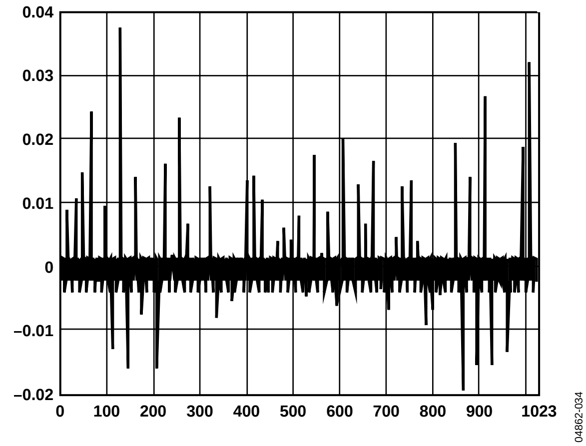


Figure 34. AD9734 DNL, 25°C, 10 mA FS

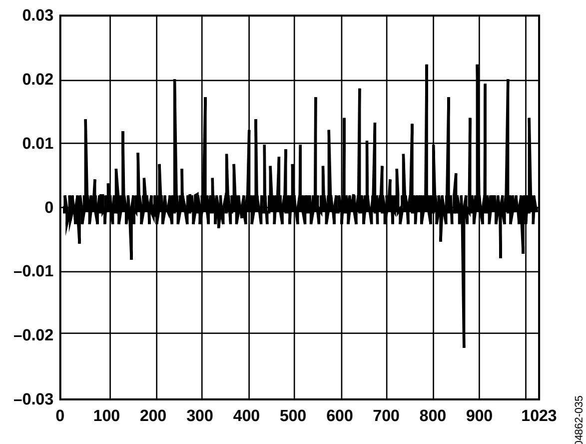


Figure 35. AD9734 DNL, 25°C, 20 mA FS

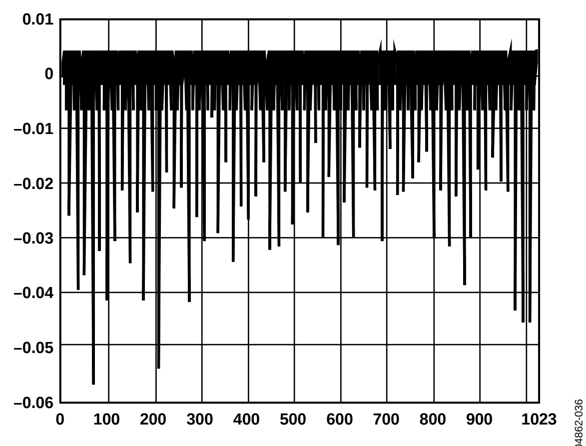


Figure 36. AD9734 DNL, 25°C, 30 mA FS

AD9736 POWER CONSUMPTION, 20 mA FULL SCALE

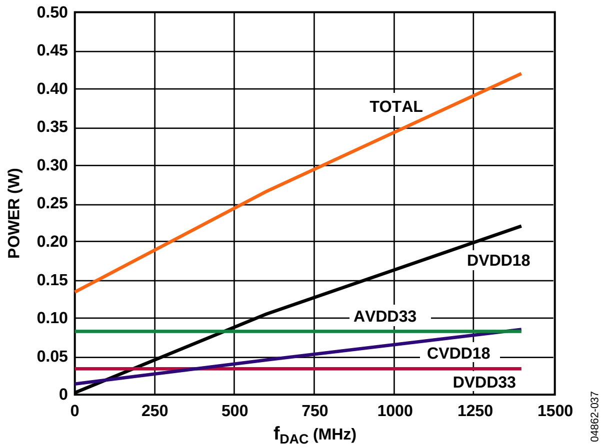


Figure 37. AD9736 1× Mode Power vs. f_{DAC} at 25°C

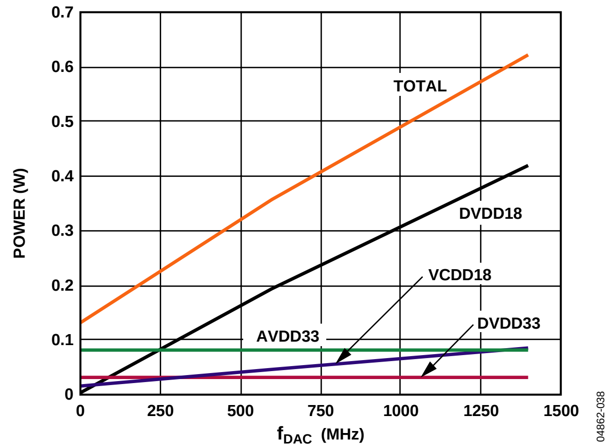


Figure 38. AD9736, 2× Interpolation Mode Power vs. f_{DAC} at 25°C

AD9736 DYNAMIC PERFORMANCE, 20 mA FULL SCALE

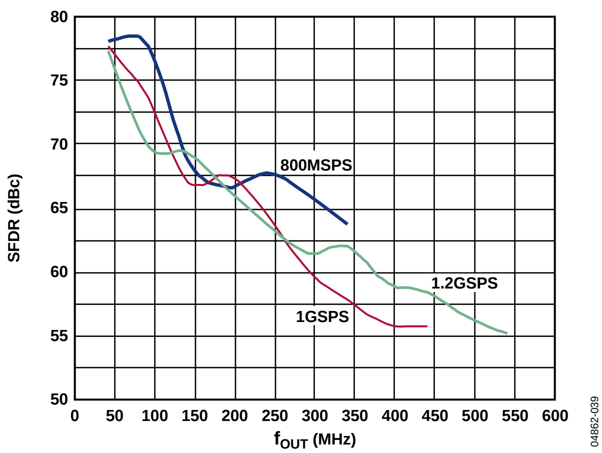


Figure 39. AD9736 SFDR vs. f_{OUT} over f_{DAC} at 25°C

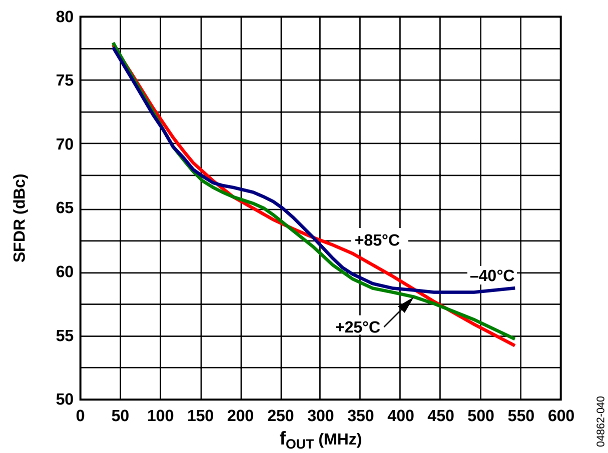


Figure 40. AD9736 SFDR vs. f_{OUT} over Temperature

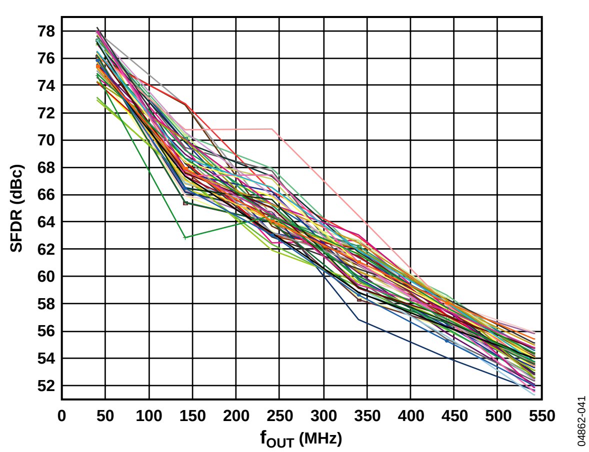


Figure 41. AD9736 SFDR vs. f_{OUT} over 50 parts, 25°C, 1.2 GSPS

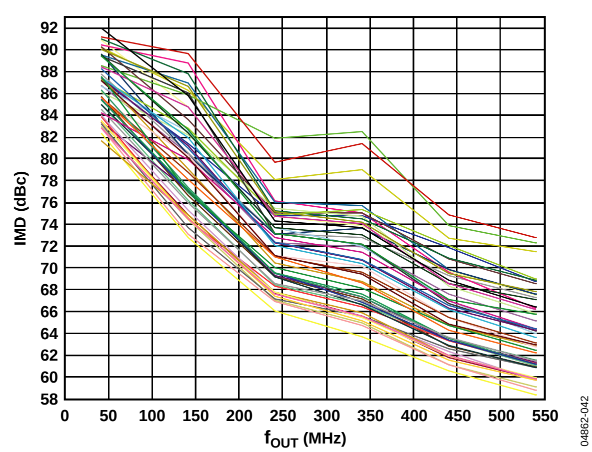


Figure 42. AD9736 IMD vs. f_{OUT} over 50 parts, 25°C,1.2 GSPS

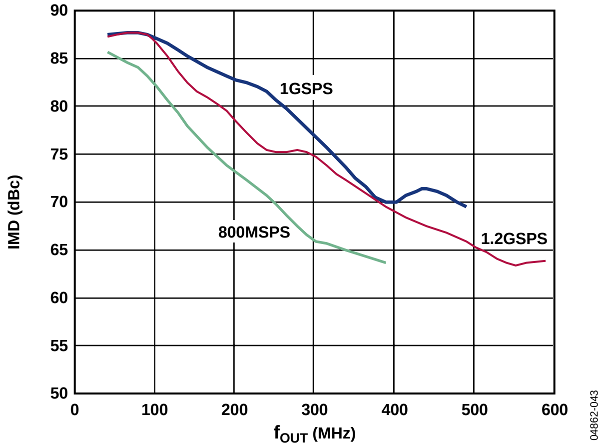


Figure 43. AD9736 IMD vs. f_{OUT} over f_{DAC} at 25°C

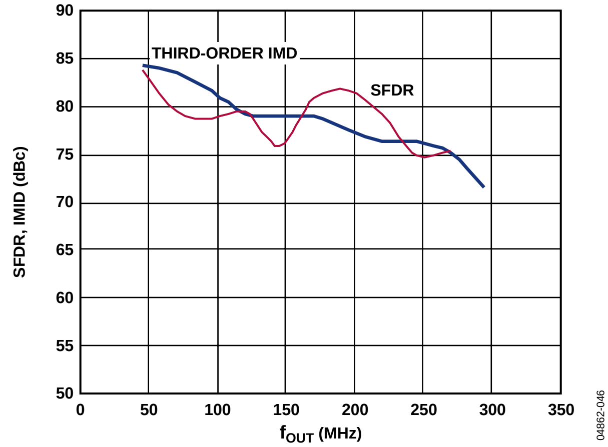


Figure 46. AD9736 IMD and SFDR vs. f_{OUT}, 25°C, 1.2 GSPS, 2× Interpolation

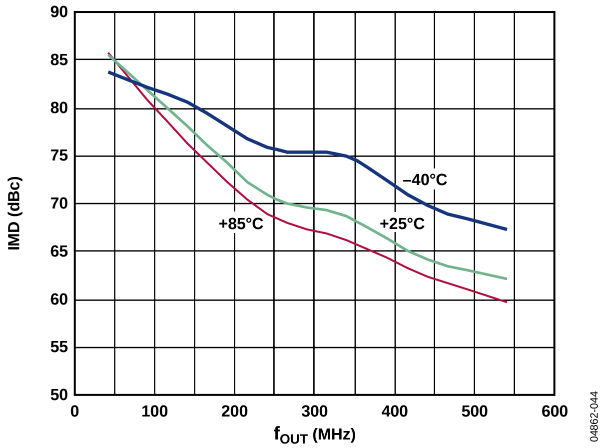


Figure 44. AD9736 IMD vs. f_{OUT} over Temperature, 1.2 GSPS

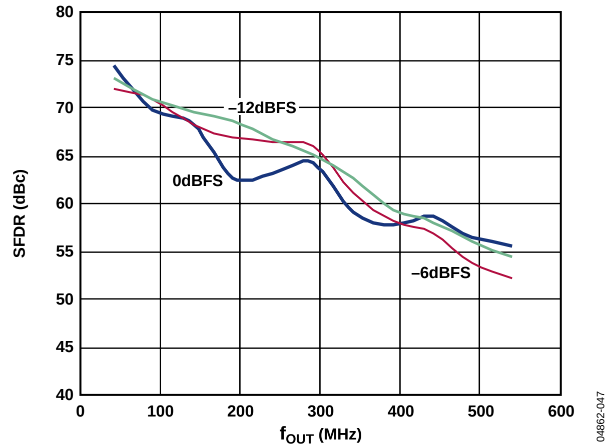


Figure 47. AD9736 SFDR vs. f_{OUT} over A_{OUT}, 25°C, 1.2 GSPS

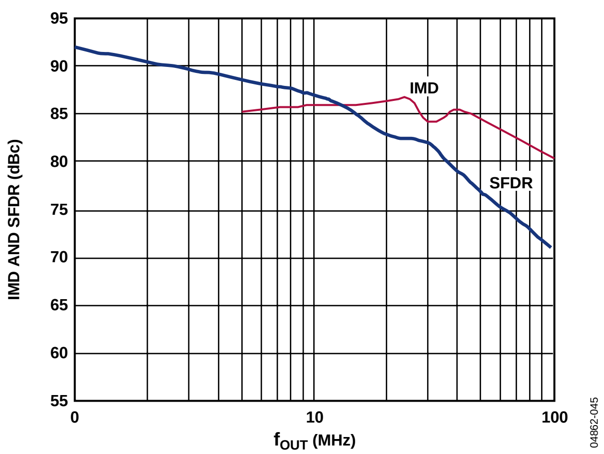


Figure 45. AD9736 Low Frequency IMD and SFDR vs. f_{OUT}, 25°C, 1.2 GSPS

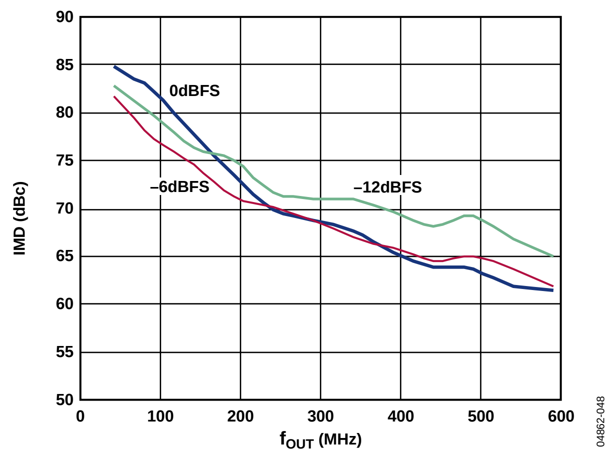


Figure 48. AD9736 IMD vs. f_{OUT} over A_{OUT}, 25°C, 1.2 GSPS

AD9736 DYNAMIC PERFORMANCE, 20 mA FULL SCALE

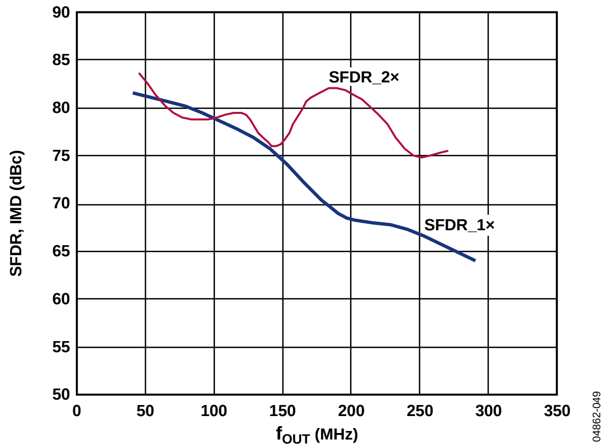


Figure 49. AD9736 SFDR vs. f_{OUT}, 25°C, 1.2 GSPS, 1× and 2× Interpolation

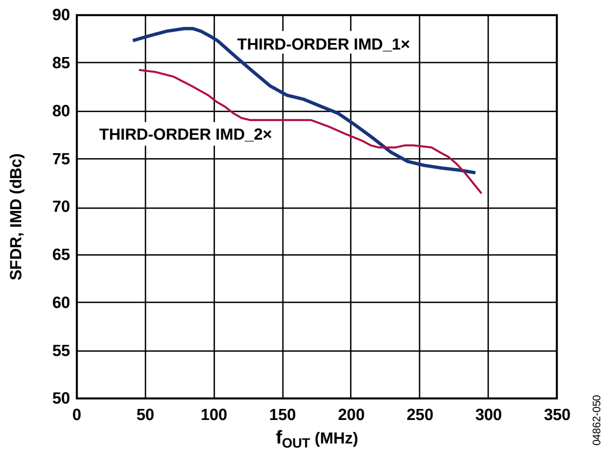


Figure 50. AD9736 IMD vs. f_{OUT}, 25°C, 1.2 GSPS, 1× and 2× Interpolation

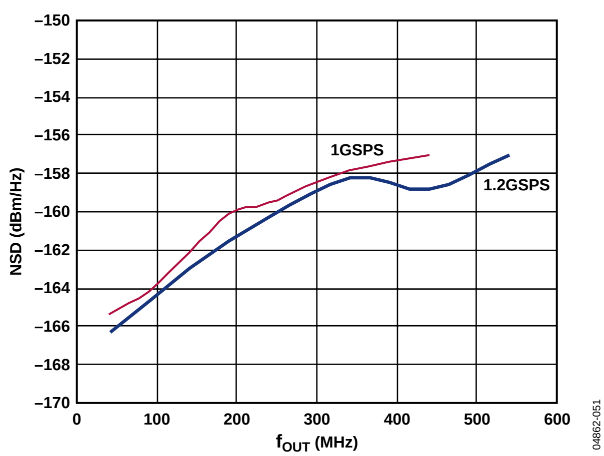


Figure 51. AD9736 1-Tone NSD vs. f_{OUT} over f_{DAC}, 25°C

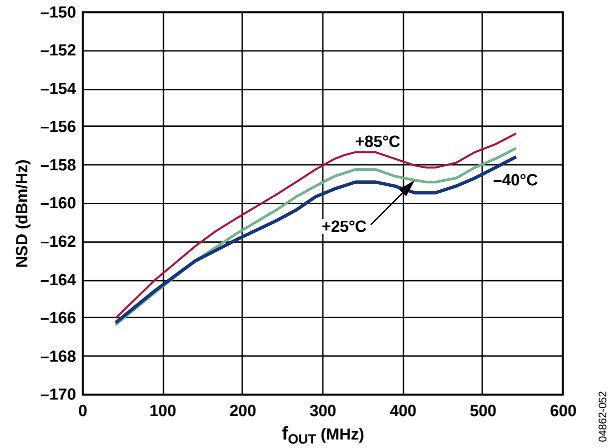


Figure 52. AD9736 1-Tone NSD vs. f_{OUT} over Temperature, 1.2 GSPS

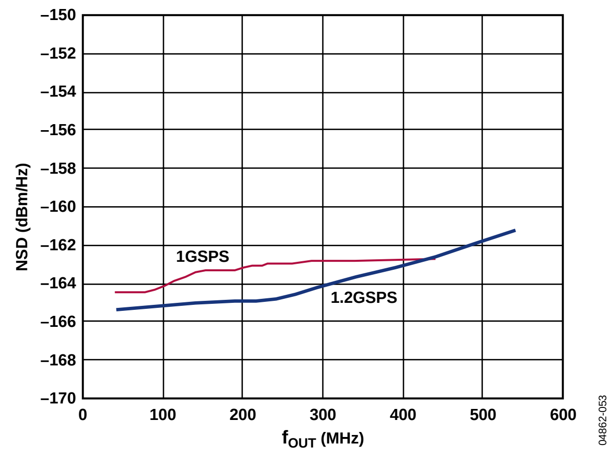


Figure 53. AD9736 8-Tone NSD vs. f_{OUT} over f_{DAC}, 25°C

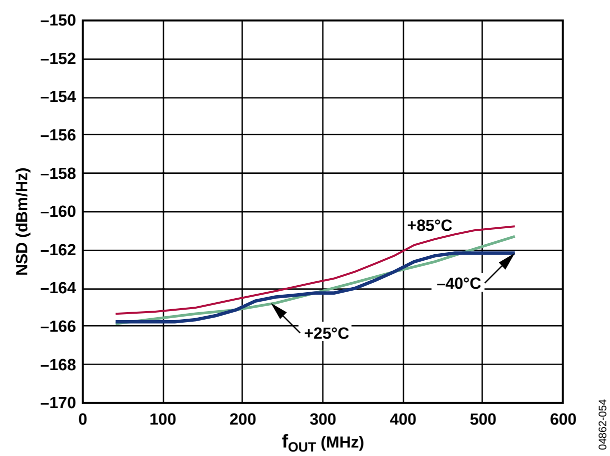


Figure 54. AD9736 8-Tone NSD vs. f_{OUT} over Temperature, 1.2 GSPS

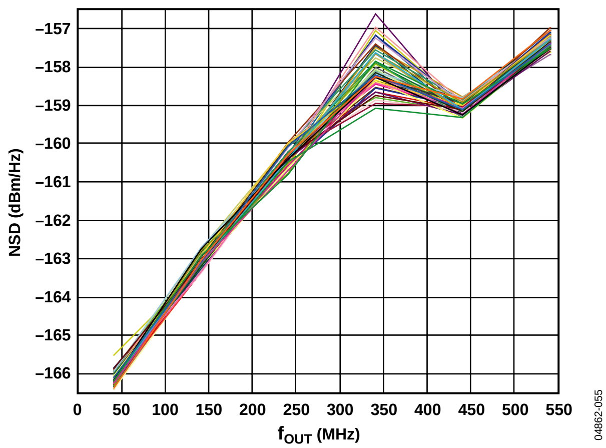


Figure 55. AD9736 1-Tone NSD vs. f_{OUT} over 50 Parts, 1.2 GSPS, 25°C

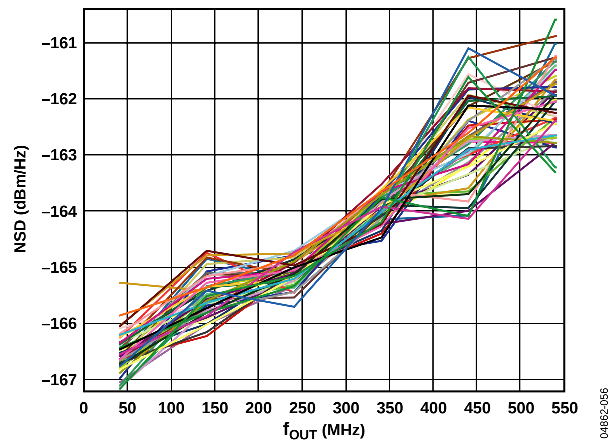


Figure 56. AD9736 1-Tone NSD vs. f_{OUT} over 50 Parts, 1.2 GSPS, 25°C

AD9736, AD9735, AD9734 WCDMA ACLR, 20 mA FULL SCALE

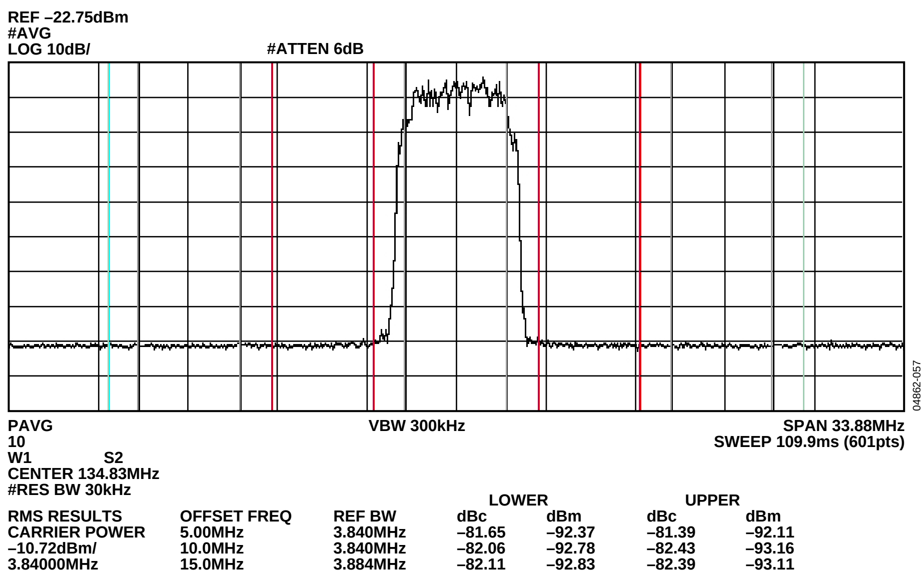
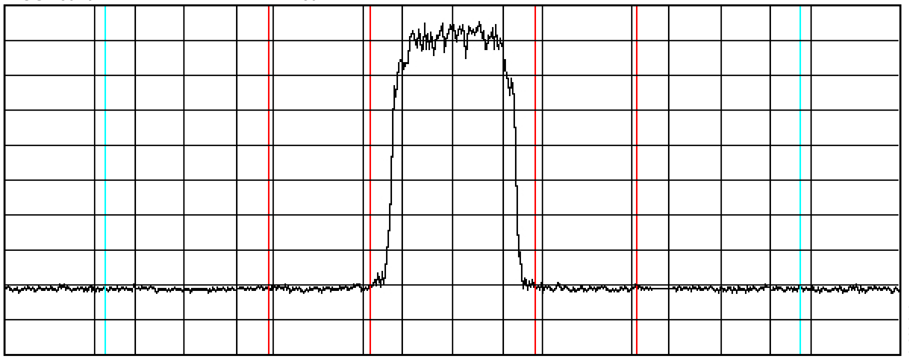
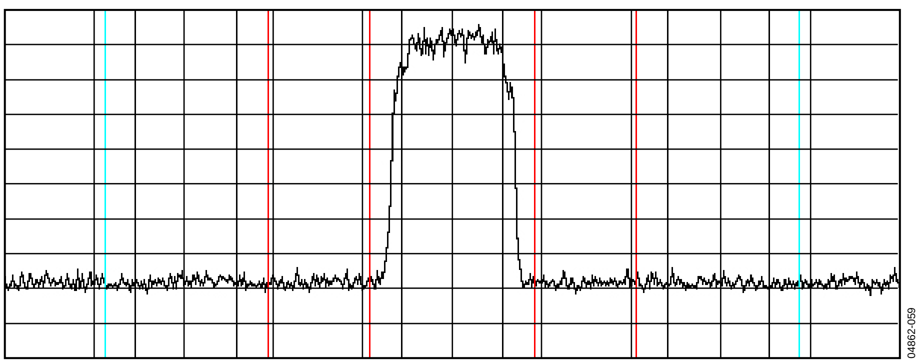


Figure 57. AD9736 WCDMA Carrier at 134.83 MHz, f_{DAC} = 491.52 MSPS



			LOWER		UPPER	
RMS RESULTS	OFFSET FREQ	REF BW	dBc	dBm	dBc	dBm
CARRIER POWER	5.00MHz	3.840MHz	–80.32	–91.10	–80.60	–91.38
–10.72dBm/	10.0MHz	3.840MHz	–81.13	–91.91	–80.75	–91.53
3.84000MHz	15.0MHz	3.884MHz	–80.43	–91.21	–81.36	–92.13

Figure 58. AD9735 WCDMA Carrier at 134.83 MHz, f_{DAC} = 491.52 MSPS



			LOWER		UPPER	
RMS RESULTS	OFFSET FREQ	REF BW	dBc	dBm	dBc	dBm
CARRIER POWER	5.00MHz	3.840MHz	–71.07	–81.83	–71.23	–81.99
–10.76dBm/	10.0MHz	3.840MHz	–70.55	–81.31	–71.42	–82.19
3.84000MHz	15.0MHz	3.884MHz	–70.79	–81.56	–71.25	–82.01

Figure 59. AD9734 WCDMA Carrier at 134.83 MHz, f_{DAC} = 491.52 MSPS

AD9735, AD9734 DYNAMIC PERFORMANCE, 20 mA FULL SCALE

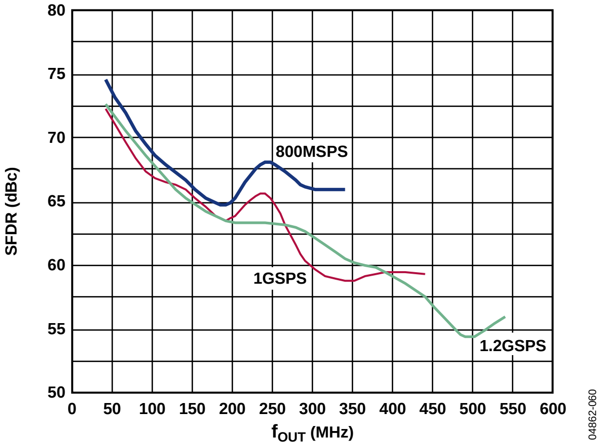


Figure 60. AD9735 SFDR vs. f_{OUT} over f_{DAC}, 1.2 GSPS

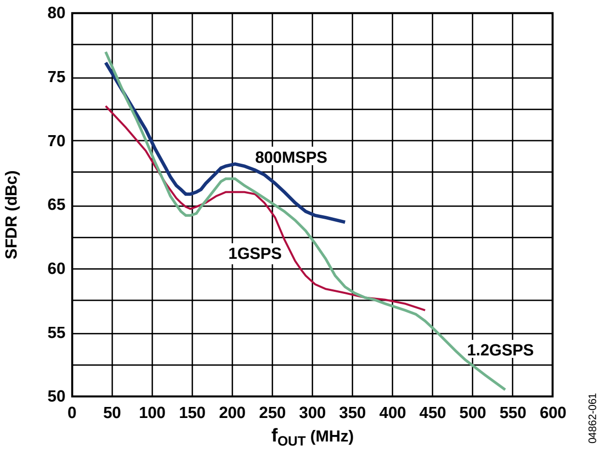


Figure 61. AD9734 SFDR vs. f_{OUT} over f_{DAC}, 1.2 GSPS

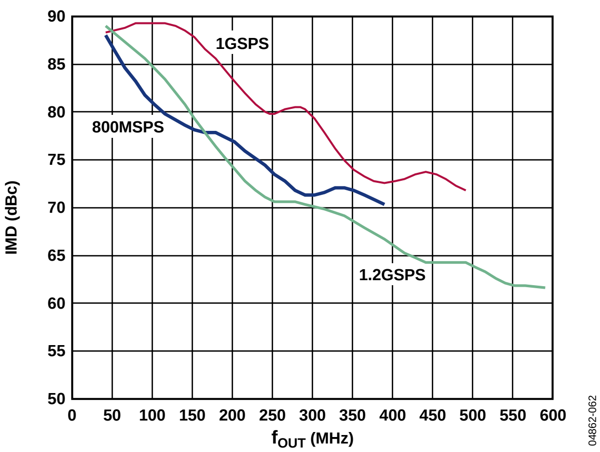


Figure 62. AD9735 IMD vs. f_{OUT} over f_{DAC}, 1.2 GSPS

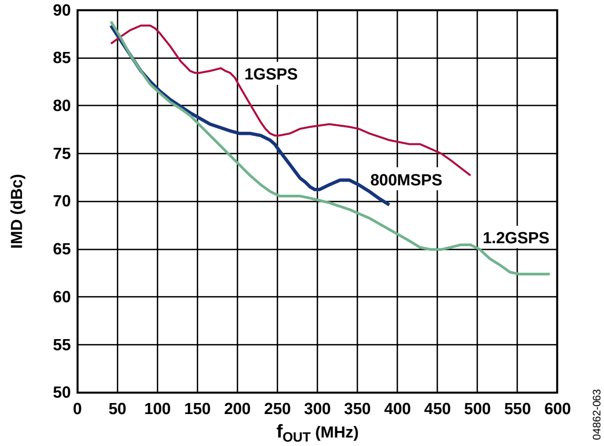


Figure 63. AD9734 IMD vs. f_{OUT} over f_{DAC}, 1.2 GSPS

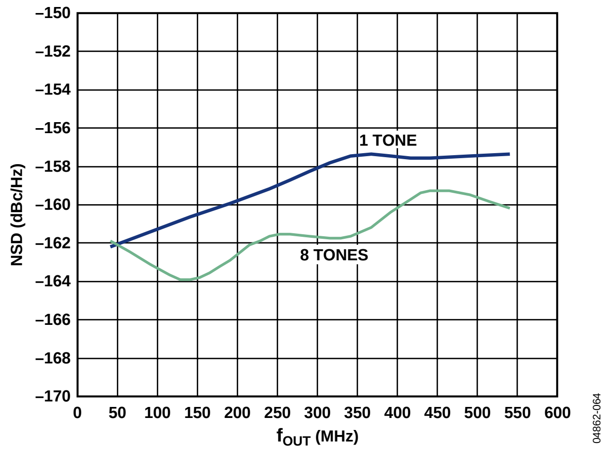


Figure 64. AD9735 NSD vs. f_{OUT}, 1.2 GSPS

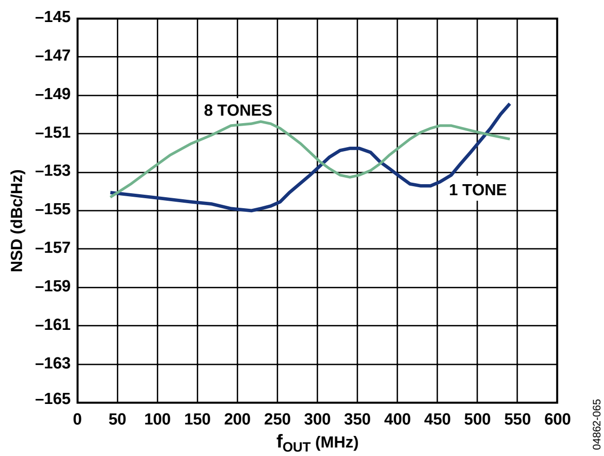


Figure 65. AD9734 NSD vs. f_{OUT}, 1.2 GSPS

SPI REGISTER MAP

Write 0 to unspecified or reserved bit locations. Reading these bits returns unknown values.

Table 6. SPI Register Map

ADR Dec	ADR Hex	Register Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Default Hex	Pin Mode Hex
0	00	MODE	SDIO_DIR	LSBFIRST	RESET	LONG_INS	2X MODE	FIFO MODE	DATAFRMT	PD	00	00
1	01	IRQ	LVDS	SYNC	CROSS	RESV'D	IE_LVDS	IE_SYNC	IE_CROSS	RESV'D	00	00
2	02	FSC_1	SLEEP						FSC<9>	FSC<8>	02	02
3	03	FSC_2	FSC<7>	FSC<6>	FSC<5>	FSC<4>	FSC<3>	FSC<2>	FSC<1>	FSC<0>	00	00
4	04	LVDS_CNT1	MSD<3>	MSD<2>	MSD<1>	MSD<0>	MHD<3>	MHD<2>	MHD<1>	MHD<0>	00	00
5	05	LVDS_CNT2	SD<3>	SD<2>	SD<1>	SD<0>	LCHANGE	ERR_HI	ERR_LO	CHECK	00	00
6	06	LVDS_CNT3	LSURV	LAUTO	LFLT<3>	LFLT<2>	LFLT<1>	LFLT<0>	LTRH<1>	LTRH<0>	00	00
7	07	SYNC_CNT1	FIFOSTAT3	FIFOSTAT2	FIFOSTAT1	FIFOSTAT0	VALID	SCHANGE	PHOF<1>	PHOF<0>	00	00
8	08	SYNC_CNT2	SSURV	SAUTO	SFLT<3>	SFLT<2>	SFLT<1>	SFLT<0>	RESV'D	STRH<0>	00	00
9	09	RESERVED										
10	0A	CROS_CNT1			UPDEL<5>	UPDEL<4>	UPDEL<3>	UPDEL<2>	UPDEL<1>	UPDEL<0>	00	00
11	0B	CROS_CNT2			DNDEL<5>	DNDEL<4>	DNDEL<3>	DNDEL<2>	DNDEL<1>	DNDEL<0>	00	00
12	0C	RESERVED										
13	0D	RESERVED										
14	0E	ANA_CNT1	MSEL<1>	MSEL<0>				TRMBG<2>	TRMBG<1>	TRMBG<0>	C0	C0
15	0F	ANA_CNT2	HDRM<7>	HDRM<6>	HDRM<5>	HDRM<4>	HDRM<3>	HDRM<2>	HDRM<1>	HDRM<0>	CA	CA
16	10	RESERVED										
17	11	BIST_CNT	SEL<1>	SEL<0>	SIG_READ			LVDS_EN	SYNC_EN	CLEAR	00	00
18	12	BIST<7:0>										
19	13	BIST<15:8>										
20	14	BIST<23:16>										
21	15	BIST<31:24>										
22	16	CCLK_DIV	RESV'D	RESV'D	RESV'D	RESV'D	CCD<3>	CCD<2>	CCD<1>	CCD<0>	00	00

SPI REGISTER DESCRIPTIONS

Reading these registers returns previously written values for all defined register bits, unless otherwise noted. Reset value for write registers in **bold** text.

MODE REGISTER (REG 00)

ADR	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x00	MODE	SDIO_DIR	LSB/MSB	RESET	LONG_INS	2× MODE	FIFO MODE	DATAFRMT	PD

Table 7. MODE Register Bit Descriptions

Bit Name	Read/Write	Description
SDIO_DIR	: WRITE ->	**0,** Input only per SPI standard 1, Bidirectional per SPI standard
LSBFIRST	: WRITE ->	**0,** MSB first per SPI standard 1, LSB first per SPI standard NOTE: Only change LSB/MSB order in single-byte instructions to avoid erratic behavior due to bit order errors.
RESET	: WRITE->	**0,** Execute software reset of SPI and controllers, reload default register values except registers 0x00 and 0x04 1, Set software reset, write 0 on the next (or any following) cycle to release the reset
LONG_INS	: WRITE ->	**0,** Short (single-byte) instruction word 1, Long (two-byte) instruction word, not necessary since the maximum internal address is REG31 (0x1F)
2×_MODE	: WRITE ->	**0,** Disable 2× interpolation filter 1, Enable 2× interpolation filter
FIFO_MODE	: WRITE ->	**0,** Disable FIFO synchronization 1, Enable FIFO synchronization
DATAFRMT	: WRITE ->	**0,** Signed input DATA with midscale = 0x0000 1, Unsigned input DATA with midscale = 0x2000
PD	: WRITE ->	**0,** Enable LVDS Receiver, DAC, and clock circuitry 1, Power down LVDS Receiver, DAC, and clock circuitry

INTERRUPT REQUEST REGISTER (IRQ) (REG 01)

ADR	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x01	IRQ	LVDS	SYNC	CROSS	RESV'D	IE_LVDS	IE_SYNC	IE_CROSS	RESV'D

Table 8. Interrupt Register Bit Descriptions

Bit Name	Read/Write	Description
LVDS	: WRITE ->	Don't Care
	: READ ->	0, No active LVDS receiver interrupt 1, Interrupt in LVDS receiver occurred
SYNC	: WRITE ->	Don't Care
	: READ ->	0, No active SYNC logic interrupt 1, Interrupt in SYNC logic occurred
CROSS	: WRITE ->	Don't Care
	: READ ->	0, No active CROSS logic interrupt 1, Interrupt in CROSS logic occurred
IE_LVDS	: WRITE ->	**0,** Reset LVDS receiver interrupt and disable future LVDS receiver interrupts 1, Enable LVDS receiver interrupt to activate IRQ pin
IE_SYNC	: WRITE ->	**0,** Reset SYNC logic interrupt and disable future SYNC logic interrupts 1, Enable SYNC logic interrupt to activate IRQ pin
IE_CROSS :	WRITE ->	**0,** Reset CROSS logic interrupt and disable future CROSS logic interrupts 1, Enable CROSS logic interrupt to activate IRQ pin

FULL SCALE CURRENT (FSC) REGISTER (REGS 02, 03)

ADR	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x02	FSC_1	SLEEP						FSC<9>	FSC<8>
0x03	FSC_2	FSC<7>	FSC<6>	FSC<5>	FSC<4>	FSC<3>	FSC<2>	FSC<1>	FSC<0>

Table 9. Full Scale Output Register Bit Descriptions

Bit Name	Read/Write	Description
SLEEP	: WRITE ->	**0,** Enable DAC output 1, Set DAC output current to 0 mA
FSC<9:0>	: WRITE ->	0x000, 10 mA full-scale output current **0x200,** 20 mA full-scale output current 0x3FF, 30 mA full-scale output current

LVDS CONTROLLER (LVDS_CNT) REGISTER (REGS 04, 05, 06)

ADR	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x04	LVDS_CNT1	MSD<3>	MSD<2>	MSD<1>	MSD<0>	MHD<3>	MHD<2>	MHD<1>	MHD<0>
0x05	LVDS_CNT2	SD<3>	SD<2>	SD<1>	SD<0>	LCHANGE	ERR_HI	ERR_LO	CHECK
0x06	LVDS_CNT3	LSURV	LAUTO	LFLT<3>	LFLT<2>	LFLT<1>	LFLT<0>	LTRH<1>	LTRH<0>

Table 10. LVDS Controller Register Bit Descriptions

Bit Name	Read/Write	Description
MSD<3:0>	: WRITE ->	**0x0,** Set setup delay for the measurement system
	: READ ->	If (LAUTO = 1) the latest measured value for the setup delay If (LAUTO = 0) read back of the last SPI write to this bit
MHD<3:0>	: WRITE ->	**0x0,** Set hold delay for the measurement system
	: READ ->	If (LAUTO = 1) the latest measured value for the hold delay If (LAUTO = 0) read back of the last SPI write to this bit
SD<3:0>	: WRITE->	**0x0,** Set sample delay
	: READ ->	If (LAUTO = 1) the result of a measurement cycle is stored in this register If (LAUTO = 0) read back of the last SPI write to this bit
LCHANGE	: READ ->	0, No change from previous measurement 1, Change in value from the previous measurement NOTE: The average filter and the threshold detection are not applied to this bit
ERR_HI	: READ ->	One of the 15 LVDS inputs is above the input voltage limits of the IEEE reduce link specification.
ERR_LO	: READ ->	One of the 15 LVDS inputs is below the input voltage limits of the IEEE reduced link specification.
CHECK	: READ ->	0, Phase measurement—sampling in the previous or following DATA cycle 1, Phase measurement—sampling in the correct DATA cycle
LSURV	: WRITE ->	**0,** The controller stops after completion of the current measurement cycle 1, Continuous measurements are taken and an interrupt is issued if the clock alignment drifts beyond the threshold value
LAUTO :	WRITE ->	**0,** Sample delay is not automatically updated 1, Continuously starts measurement cycles and updates the sample delay according to the measurement NOTE: LSURV (Reg 06, Bit 7) must be set to 1 and the LVDS IRQ (Reg 01 Bit 3) must be set to 0 for AUTO mode
LFLT<3:0>	: WRITE ->	**0x0,** Average filter length, Delay = Delay + Delta Delay / 2^ LFLT<3:0>, values greater than 12 (0x0C) are clipped to 12
LTRH<2:0> :	: WRITE ->	**000,** Set auto update threshold values

SYNC CONTROLLER (SYNC_CNT) REGISTER (REGS 07, 08)

ADR	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x07	SYNC_CNT1	FIFOSTAT3	FIFOSTAT2	FIFOSTAT1	FIFOSTAT0	VALID	SCHANGE	PHOF<1>	PHOF<0>
0x08	SYNC_CNT2	SSURV	SAUTO	SFLT<3>	SFLT<2>	SFLT<1>	SFLT<0>	RESV'D	STRH<0>

Table 11. SYNC Controller Register Bit Descriptions

Bit Name	Read/Write	Description
FIFOSTAT<2:0>	: READ ->	Position of FIFO read counter, range from 0 to 7
FIFOSTAT<3>	: READ ->	0, SYNC logic OK 1, Error in SYNC logic
VALID	: READ ->	0, FIFOSTAT<3:0> is not valid yet 1, FIFOSTAT<3:0> is valid after a reset
SCHANGE	: READ ->	0, No change in FIFOSTAT<3:0> 1, FIFOSTAT<3:0> has changed since the previous measurement cycle when SSURV = 1 (surveillance mode active)
PHOF<1:0>	: WRITE ->	**00,** Change the readout counter
	: READ ->	Current setting of the readout counter (PHOF<1:0>) in surveillance mode (SSURV = 1) after an interrupt Current calculated optimal readout counter value in AUTO mode (SAUTO = 1)
SSURV :	WRITE ->	**0,** The controller stops after completion of the current measurement cycle 1, Continuous measurements are taken and an interrupt is issued if the readout counter drifts beyond the threshold value
SAUTO	: WRITE ->	**0,** Readout counter (PHOF<3:0>) is not automatically updated 1, Continuously starts measurement cycles and updates the readout counter according to the measurement NOTE: SSURV (Reg 08 Bit 7) must be set to 1 and the SYNC IRQ (Reg 01 Bit 2) must be set to 0 for AUTO mode
SFLT<3:0>	: WRITE ->	**0x0,** Average filter length, FIFOSTAT = FIFOSTAT + Delta FIFOSTAT/2 ^ SFLT<3:0>, values greater than 12 (0x0C) are clipped to 12
STRH<0>	: WRITE ->	**0,** If FIFOSTAT<2:0> = 0 | 7, generate a SYNC interrupt 1, If FIFOSTAT<2:0> = 0 | 1 | 6 | 7, generate a SYNC interrupt

CROSS CONTROLLER (CROS_CNT) REGISTER (REGS 10, 11)

ADR	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x0A	CROS_CNT1	–	–	UPDEL<5>	UPDEL<4>	UPDEL<3>	UPDEL<2>	UPDEL<1>	UPDEL<0>
0x0B	CROS_CNT2	–	–	DNDEL<5>	DNDEL<4>	DNDEL<3>	DNDEL<2>	DNDEL<1>	DNDEL<0>

Table 12. Cross Controller Register Description

Bit Name	Read/Write	Description
UPDEL<5:0>	: WRITE ->	**0x00,** Move the differential output stage switching point up, set to 0 if DNDEL is non-zero
DNDEL<5:0>	: WRITE ->	**0x00,** Move the differential output stage switching point down, set to 0 if UPDEL is non-zero

ANALOG CONTROL (ANA_CNT) REGISTER (REGS 14, 15)

ADR	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x0E	ANA_CNT1	MSEL<1>	MSEL<0>	–	–	–	TRMBG<2>	TRMBG<1>	TRMBG<0>
0x0F	ANA_CNT2	HDRM<7>	HDRM<6>	HDRM<5>	HDRM<4>	HDRM<3>	HDRM<2>	HDRM<1>	HDRM<0>

Table 13. Analog Control Register Bit Descriptions

Bit Name	Read/Write	Description
MSEL<1:0>	: WRITE ->	00, Mirror roll off frequency control = bypass 01, Mirror roll off frequency control = narrowest bandwidth 10, Mirror roll off frequency control = medium bandwidth **11,** Mirror roll off frequency control = widest bandwidth NOTE: See plot in the Analog Control Registers section.
TRMBG<2:0>	: WRITE ->	**000,** Band gap temperature characteristic trim NOTE: See the plot in the Analog Control Registers section.
HDRM<7:0>	: WRITE ->	**0xCA,** Output stack headroom control HDRM<7:4> set reference offset from AVDD33 (VCAS centering) HDRM<3:0> set overdrive (current density) trim (temperature tracking) Note: Set to 0xCA for optimum performance

BUILT-IN SELF TEST CONTROL (BIST_CNT) REGISTERS (REGS 17, 18, 19, 20, 21)

ADR	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x11	BIST_CNT	SEL<1>	SEL<0>	SIG_READ			LVDS_EN	SYNC_EN	CLEAR
0x12	BIST<7:0>	BIST<7>	BIST<6>	BIST<5>	BIST<4>	BIST<3>	BIST<2>	BIST<1>	BIST<0>
0x13	BIST<15:8>	BIST<15>	BIST<14>	BIST<13>	BIST<12>	BIST<11>	BIST<10>	BIST<9>	BIST<8>
0x14	BIST<23:16>	BIST<23>	BIST<22>	BIST<21>	BIST<20>	BIST<19>	BIST<18>	BIST<17>	BIST<16>
0x15	BIST<31:24>	BIST<31>	BIST<30>	BIST<29>	BIST<28>	BIST<27>	BIST<26>	BIST<25>	BIST<24>

Table 14. BIST Register Bit Descriptions

Bit Name	Read/Write	Description
SEL<1:0>	: WRITE ->	**00,** Write result of the LVDS Phase 1 BIST to BIST<31:0> 01, Write result of the LVDS Phase 2 BIST to BIST<31:0> 10, Write result of the SYNC Phase 1 BIST to BIST<31:0> 11, Write result of the SYNC Phase 2 BIST to BIST<31:0>
SIG_READ	: WRITE ->	**0,** No action 1, Enable BIST signature readback
LVDS_EN	WRITE->	**0,** No action 1, Enable LVDS BIST
SYNC_EN	: WRITE ->	**0,** No action 1, Enable SYNC BIST
CLEAR	: WRITE ->	**0,** No action 1, Clear all BIST registers
BIST<31:0>	: READ ->	Results of the built-in self test

CONTROLLER CLOCK PREDIVIDER (CCLK_DIV) READING REGISTER (REG 22)

ADR	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x16	CCLK_DIV	RESV'D	RESV'D	RESV'D	RESV'D	CCD<3>	CCD<2>	CCD<1>	CCD<0>

Table 15. Clock Predivider Register Bit Descriptions

Bit Name	Read/Write	Description
CCD<3:0>	: WRITE ->	**0x0,** Controller clock = DACCLK/16 0x1, Controller clock = DACCLK/32 0x2, Controller clock = DACCLK/64 … 0xF, Controller clock = DACCLK/524288 NOTE: The 100 MHz to 1.2 GHz DACCLK must be divided to less than 10 MHz for correct operation. CCD<3:0> must be programmed to divide the DACCLK so that this relationship is not violated. Controller Clock = DACCLK/(2 ^ (CCD<3:0> + 4))

THEORY OF OPERATION

The AD9736, AD9735, and AD9734 are 14-, 12-, and 10-bit DACs that run at an update rate up to 1.2 GSPS. Input data can be accepted up to the full 1.2 GSPS rate, or a 2× interpolation filter may be enabled (2× mode) allowing full speed operation with a 600 MSPS input data rate. DATA and DATACLK_IN inputs are parallel LVDS, meeting the IEEE reduced swing LVDS specifications with the exception of input hysteresis. The DATACLK_IN input runs at one-half the input DATA rate in a double data rate (DDR) format. Each edge of DATACLK_IN is used to transfer DATA into the AD9736, as shown in Figure 77.

The DACCLK−/DACCLK+ inputs (Pins E1, F1) directly drive the DAC core to minimize clock jitter. The DACCLK signal is also divided by 2 (1× and 2× mode), then output as the DATACLK_OUT. The DATACLK_OUT signal is used to clock the data source. The DAC expects DDR LVDS data (DB<13:0>) aligned with the DDR input clock (DATACLK_IN) from a circuit similar to the one shown in Figure 94. Table 16 shows the clock relationships.

Table 16. AD973x Clock Relationship

MODE	DACCLK	DATACLK_OUT	DATACLK_IN	DATA
1×	1.2 GHz	600 MHz	600 MHz	1.2 GSPS
2×	1.2 GHz	600 MHz	300 MHz	600 MSPS

Maintaining correct alignment of data and clock is a common challenge with high speed DACs, complicated by changes in temperature and other operating conditions. Use of the DATACLK_OUT signal to generate the data allows most of the internal process, temperature, and voltage delay variation to be cancelled. The AD973x further simplifies this high speed data capture problem with two adaptive closed-loop timing controllers.

One timing controller manages the LVDS data and data clock alignment (LVDS controller), and the other manages the LVDS data and DACCLK alignment (SYNC controller). The LVDS controller locates the data transitions and delays the DATACLK_IN so that its transition is in the center of the valid data window. The SYNC controller manages the FIFO that moves data from the LVDS DATACLK_IN domain to the DACCLK domain. Both controllers can be operated in manual mode under external processor control, surveillance mode where error conditions generate external interrupts, or automatic mode where errors are automatically corrected.

The LVDS and SYNC controllers include moving average filtering for noise immunity and variable thresholds to control their activity. Normally the controllers can be set to run in automatic mode, and they make any necessary adjustments without dropping or duplicating samples sent to the DAC. Both controllers require initial calibration prior to entering automatic update mode.

The AD973x analog output changes 35 DACCLK cycles after the input data changes in 1× mode with the FIFO disabled. The FIFO can add up to eight additional cycles of delay. This delay can be read from the SPI port. Internal clock delay variation is less than a single DACCLK cycle at 1.2 GHz (833 ps).

Stopping the AD973x DATACLK_IN while the DACCLK is still running can lead to unpredictable output signals. This occurs because the internal digital signal path is interleaved. The last two samples clocked into the DAC continue to be clocked out by DACCLK even after DATACLK_IN has been stopped. The resulting output signal is at a frequency of one-half f_{DAC}, and the amplitude depends on the difference between the last two samples.

Control of the AD973x functions is via the serially programmed registers listed in Table 6. Optionally, a limited number of functions may be directly set by external pins in pin mode.

SERIAL PERIPHERAL INTERFACE

The AD973x serial port is a flexible, synchronous serial communications port, allowing easy interface to many industry-standard microcontrollers and microprocessors. The serial I/O is compatible with most synchronous transfer formats, including both the Motorola SPI® and Intel® SSR protocols. The interface allows read/write access to all registers that configure the AD973x. Single- or multiple-byte transfers are supported, as well as most significant bit first (MSB-first) or least significant bit first (LSB-first) transfer formats. The AD973x serial interface port can be configured as a single pin I/O (SDIO) or two unidirectional pins for in/out (SDIO/SDO).

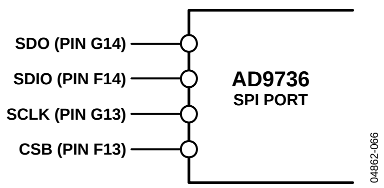


Figure 66. AD973x SPI Port

The AD973x can optionally be configured via external pins rather than the serial interface. When the PIN_MODE input (Pin L1) is high, the serial interface is disabled and its pins are reassigned for direct control of the DAC. Specific functionality is described in the Pin Mode Operation section.

GENERAL OPERATION OF THE SERIAL INTERFACE

There are two phases to a communication cycle with the AD973x. Phase 1 is the instruction cycle, which is the writing of an instruction byte into the AD973x, coincident with the first eight SCLK rising edges. The instruction byte provides the AD973x serial port controller with information regarding the data transfer cycle, which is Phase 2 of the communication cycle. The Phase 1 instruction byte defines whether the upcoming data transfer is read or write, the number of bytes in the data transfer, and the starting register address for the first byte of the data transfer. The first eight SCLK rising edges of each communication cycle are used to write the instruction byte into the AD973x.

The remaining SCLK edges are for Phase 2 of the communication cycle. Phase 2 is the actual data transfer between the AD973x and the system controller. Phase 2 of the communication cycle is a transfer of 1, 2, 3, or 4 data bytes as determined by the instruction byte. Using one multibyte transfer is the preferred method. Single-byte data transfers are useful to reduce CPU overhead when register access requires one byte only. Registers change immediately upon writing to the last bit of each transfer byte.

CSB (Chip Select) can be raised after each sequence of 8 bits (except the last byte) to stall the bus. The serial transfer resumes when CSB is lowered. Stalling on nonbyte boundaries resets the SPI.

SHORT INSTRUCTION MODE (8-BIT INSTRUCTION)

The short instruction byte is shown in Table 17.

Table 17. SPI Instruction Byte

MSB							LSB
I7	I6	I5	I4	I3	I2	I1	I0
R/W	N1	N0	A4	A3	A2	A1	A0

R/W, Bit 7 of the instruction byte, determines whether a read or a write data transfer occurs after the instruction byte write. Logic high indicates read operation. Logic 0 indicates a write operation. N1, N0, Bits 6 and 5 of the instruction byte, determine the number of bytes to be transferred during the data transfer cycle. The bit decodes are shown in Table 18.

A4, A3, A2, A1, A0, Bits 4, 3, 2, 1, 0 of the instruction byte, determine which register is accessed during the data transfer portion of the communications cycle. For multibyte transfers, this address is the starting byte address. The remaining register addresses are generated by the AD973x, based on the LSBFIRST bit (Reg 00, Bit 6).

Table 18. Byte Transfer Count

N1	N2	Description
0	0	Transfer 1 byte
0	1	Transfer 2 bytes
1	0	Transfer 3 bytes
1	1	Transfer 4 bytes

LONG INSTRUCTION MODE (16-BIT INSTRUCTION)

The long instruction bytes are shown in Table 19.

Table 19. SPI Instruction Byte

MSB							LSB
I15 R/W	I14 N1	I13 N0	I12 A12	I11 A11	I10 A10	I9 A9	I8 A8
I7 A7	I6 A6	I5 A5	I4 A4	I3 A3	I2 A2	I1 A1	I0 A0

If LONG_INS = 1 (Reg 00, Bit 4), the instruction byte is extended to 2 bytes where the second byte provides an additional 8 bits of address information. Addresses 0x00 to 0x1F are equivalent in short and long instruction modes. The AD973x does not use any addresses greater than 31 (0x1F), so always set LONG_INS = 0.

SERIAL INTERFACE PORT PIN DESCRIPTIONS

SCLK—Serial Clock. The serial clock pin is used to synchronize data to and from the AD973x and to run the internal state machines. The maximum frequency of SCLK is 20 MHz. All data input to the AD973x is registered on the rising edge of SCLK. All data is driven out of the AD973x on the rising edge of SCLK.

CSB—Chip Select. Active low input starts and gates a communication cycle. It allows more than one device to be used on the same serial communications lines. The SDO and SDIO pins goes to a high impedance state when this input is high. Chip select should stay low during the entire communication cycle.

SDIO—Serial Data I/O. Data is always written into the AD973x on this pin. However, this pin can be used as a bidirectional data line. The configuration of this pin is controlled by SDIO_DIR at Reg 00, Bit 7. The default is Logic 0, which configures the SDIO pin as unidirectional.

SDO—Serial Data Out. Data is read from this pin for protocols that use separate lines for transmitting and receiving data. In the case where the AD973x operates in a single bidirectional I/O mode, this pin does not output data and is set to a high impedance state.

MSB/LSB TRANSFERS

The AD973x serial port can support both MSB-first or LSB-first data formats. This functionality is controlled by LSBFIRST at Reg 00, Bit 6. The default is MSB first (LSBFIRST = 0).

When LSBFIRST = 0 (MSB first), the instruction and data bytes must be written from the most significant bit to the least significant bit. Multibyte data transfers in MSB-first format start with an instruction byte that includes the register address of the most significant data byte. Subsequent data bytes should follow in order from high address to low address. In MSB-first mode, the serial port internal byte address generator decrements for each data byte of the multibyte communication cycle.

When LSBFIRST = 1 (LSB first), the instruction and data bytes must be written from least significant bit to most significant bit. Multibyte data transfers in LSB-first format start with an instruction byte that includes the register address of the least significant data byte followed by multiple data bytes. The serial port internal byte address generator increments for each byte of the multibyte communication cycle.

The AD973x serial port controller data address decrements from the data address written toward 0x00 for multibyte I/O operations if the MSB-first mode is active. The serial port controller address increments from the data address written toward 0x1F for multibyte I/O operations if the LSB-first mode is active.

NOTES ON SERIAL PORT OPERATION

The AD973x serial port configuration is controlled by Reg 00, Bits 4, 5, 6, and 7. Note that the configuration changes immediately upon writing to the last bit of the register. For multibyte transfers, writing to this register may occur during the middle of communication cycle. Care must be taken to compensate for this new configuration for the remaining bytes of the current communication cycle. The same considerations apply to setting the software reset, RESET (Reg 00, Bit 5). All registers are set to their default values except Reg 00 and Reg 04 which remain unchanged.

Use of only single-byte transfers when changing serial port configurations or initiating a software reset is highly recommended. In the event of unexpected programming sequences, the AD973x SPI may become inaccessible. For example, if user code inadvertently changes the LONG_INS bit or the LSBFIRST bit, the following bits may have unexpected results. The SPI can be returned to a known state by writing an incomplete byte (1 to 7 bits) of all 0s followed by 3 bytes of 0x00. This returns to MSB-first short instructions (Reg 00 = 0x00) so the device may be reinitialized.

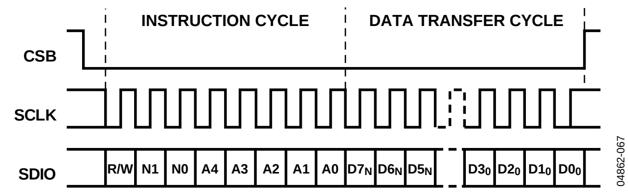


Figure 67. Serial Register Interface Timing, MSB-First Write

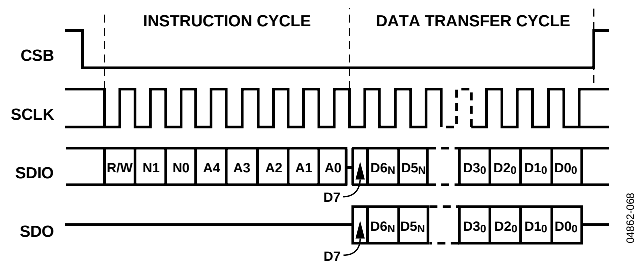


Figure 68. Serial Register Interface Timing, MSB-First Read

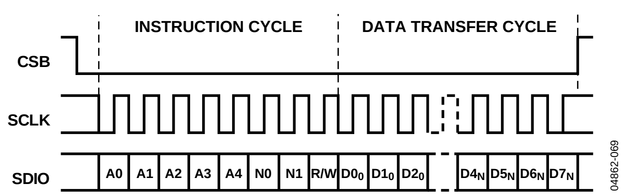


Figure 69. Serial Register Interface Timing, LSB-First Write

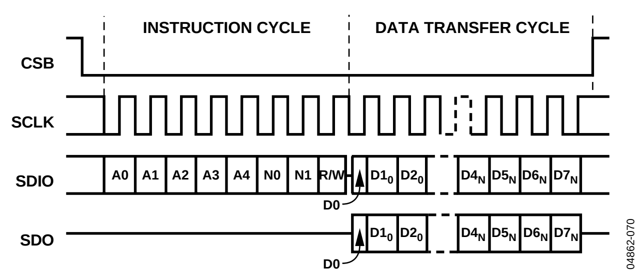


Figure 70. Serial Register Interface Timing, LSB-First Read

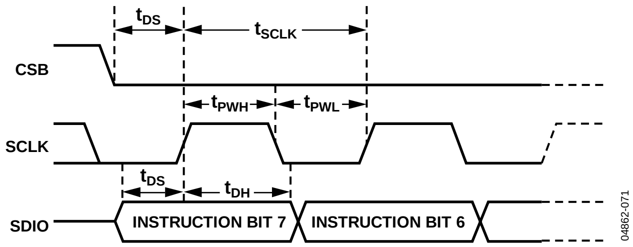


Figure 71. Timing Diagram for SPI Register Write

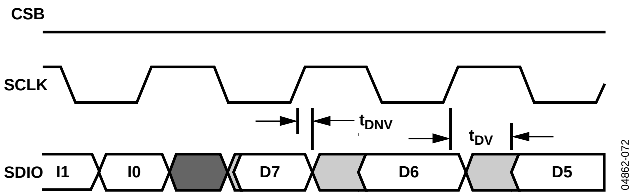


Figure 72. Timing Diagram for SPI Register Read

After the last instruction bit is written to the SDIO pin, the driving signal must be set to a high impedance in time for the bus to turn around. The serial output data from the AD973x is enabled by the falling edge of SCLK. This causes the first output data bit to be shorter than the remaining data bits, as shown in Figure 72.

To assure proper reading of data, read the SDIO or SDO pin prior to changing the SCLK from low to high.

Due to the more complex multibyte protocol, multiple AD973x devices cannot be daisy-chained on the SPI bus. Multiple DACs should be controlled by independent CSB signals.

PIN MODE OPERATION

When the PIN_MODE input (Pin L1) is set high, the SPI port is disabled. The SPI port pins are remapped, as shown in Table 20. The function of these pins is described in Table 21. The remaining PIN_MODE register settings are shown in Table 6.

Table 20. SPI_MODE vs. PIN_MODE Inputs

Pin No.	PIN_MODE = 0	PIN_MODE = 1
E13	IRQ	UNSIGNED
F13	CSB	2×
G13	SCLK	FSC0
E14	RESET	PD
F14	SDIO	FIFO
G14	SDO	FSC1

Table 21. PIN_MODE Input Functions

Pin	Function
UNSIGNED	0, Twos complement input data format
	1, Unsigned input data format
2×	0, Interpolation disabled
	1, Interpolation = 2× enabled
FSC1, FSC0	00, Sleep mode
	01, 10 mA full-scale output current
	10, 20 mA full-scale output current
	11, 30 mA full-scale output current
PD	0, Chip enabled
	1, Chip in power-down state
FIFO	0, Input FIFO disabled
	1, Input FIFO enabled

Care must be taken when using PIN_MODE, because only the control bits shown in Table 21 can be changed. If the remaining register default values are not suitable for the desired operation, PIN_MODE cannot be used. If the FIFO is enabled, the controller clock must be less than 10 MHz. This limits the DAC clock to 160 MHz.

RESET OPERATION

The RESET pin forces all SPI register contents to their default values (see Table 6), which places the DAC in a known state. The software reset bit forces all SPI register contents, except Reg 00 and Reg 04, to their default values.

The internal reset signal is derived from a logical OR operation on the RESET pin state and from the software reset state. This internal reset signal drives all SPI registers to their default values, except Reg 00 and Reg 04, which are unaffected. The data registers are not affected by either reset.

The software reset is asserted by writing 1 to Reg 00, Bit 5. It may be cleared on the next SPI write cycle or a later write cycle.

PROGRAMMING SEQUENCE

The AD973x registers should be programmed in this order:

1. Hardware reset
2. SPI port configuration changes, if necessary
3. Input format, if unsigned
4. Interpolation, if in 2× mode
5. Calibrate and set the LVDS Controller
6. Enable the FIFO
7. Calibrate and set the sync controller

Steps 1 through 4 are required, while 5 through 7 are optional. The LVDS controller can help assure proper data reception in the DAC with changes in temperature and voltage. The SYNC controller manages the FIFO to assure proper transfer of the received data to the DAC core with changes in temperature and voltage. The DAC is intended to operate with both controllers active unless data and clock alignment is managed externally.

INTERPOLATION FILTER

In 2× mode, the input data is interpolated by a factor of 2 so it aligns with the DAC update rate. The interpolation filter is a hard-coded, 55-tap, symmetric FIR with a 0.001 dB pass-band flatness and a stop-band attenuation of about 90 dB. The transition band runs from 20% of f_{DAC} to 30% of f_{DAC}. The FIR response is shown in Figure 73 where the frequency axis is normalized to f_{DAC} Figure 74 shows the pass-band flatness and Table 22 shows the 16-bit filter coefficients.

Table 22. FIR Interpolation Filter Coefficients

Coefficient No.	Coefficient No.	Tap Weight
h1	h55	−7
h2	h54	0
h3	h53	24
h4	h52	0
h5	h51	−62
h6	h50	0
h7	h49	135
h8	h48	0
h9	h47	−263
h10	h46	0
h11	h45	471
h12	h44	0
h13	h43	−793
h14	h42	0
h15	h41	1273
h16	h40	0
h17	h39	−1976
h18	h38	0
h19	h37	3012
h20	h36	0
h21	h35	−4603
h22	h34	0
h23	h33	7321
h24	h32	0
h25	h31	−13270
h26	h30	0
h27	h29	41505
h28		65536

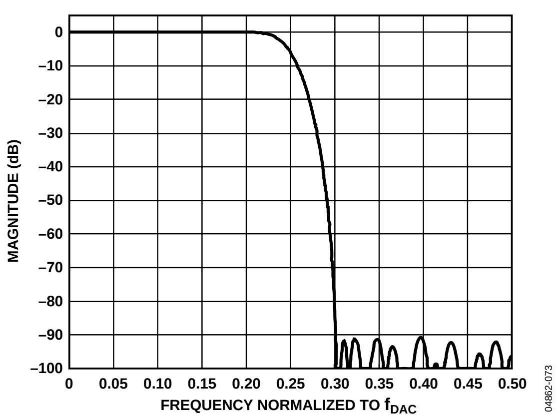


Figure 73. AD973x Interpolation Filter Response

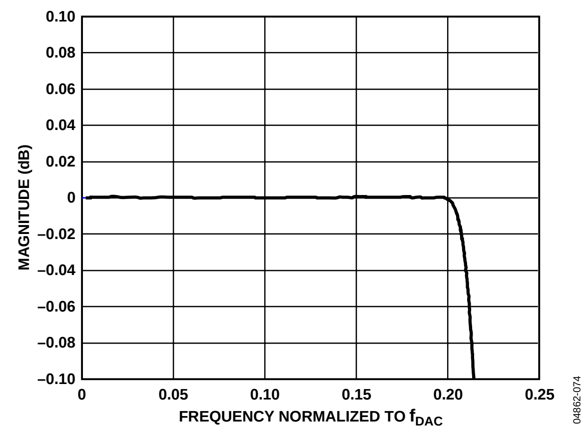


Figure 74. AD973x Interpolation Filter Pass-Band Flatness

DATA INTERFACE CONTROLLERS

There are two internal controllers that can be utilized in the operation of the AD973x. The first controller helps maintain optimum LVDS data sampling and the second controller helps maintain optimum synchronization between the DACCLK and the incoming data. The LVDS controller is responsible for optimizing the sampling of the data from the LVDS bus (DB13:0), while the SYNC controller resolves timing problems between the DAC_CLK (CLK+, CLK–) and the DATACLK. A block diagram of these controllers is shown in Figure 75.

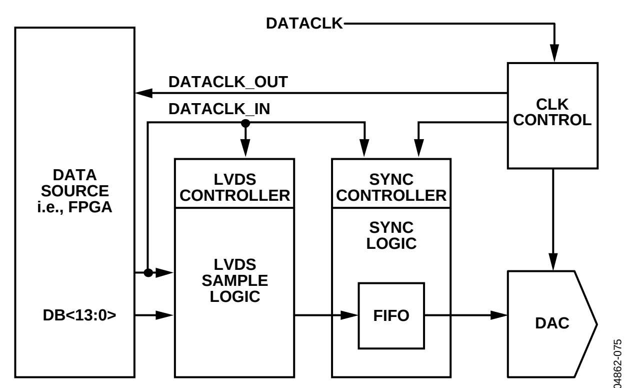


Figure 75. AD973x Data Controllers

The controllers are clocked with a divided-down version of the DAC_CLK. The divide ratio is set utilizing the controller clock predivider bits (CCD<3:0>) located at REG 22, Bits 3:0 to generate the controller clock as follows:

$$Controller\ Clock = DAC_CLK/(2(CCD<3:0> + 4))$$

Note that the controller clock may not exceed 10 MHz for correct operation. Until CCD<3:0> has been properly programmed to meet this requirement, the DAC output may not be stable. This means the FIFO cannot be enabled in PIN_MODE unless the DACCLK is less than 160 MHz.

The LVDS and SYNC controllers can be independently operated in three modes via SPI port Reg 06 and Reg 08:

- Manual mode
- Surveillance mode
- Auto mode

In manual mode, all of the timing measurements and updates are externally controlled via the SPI.

In surveillance mode, each controller takes measurements and calculates a new optimal value continuously. The result of the measurement can be passed through an averaging filter before evaluating the results for increased noise immunity. The filtered result is compared to a threshold value set via Reg 06 and Reg 08 of the SPI port. If the error is greater than the threshold, an interrupt is triggered and the controller stops. Reg 01 of the SPI port controls the interrupts with Bits 3 and 2 enabling the respective interrupts and Bits 7 and 6 indicating the respective controller's interrupt. If an interrupt is enabled, it also activates the AD973x IRQ pin. In order to clear an interrupt, the interrupt enable bit of the respective controller must be set to 0 for at least 1 controller clock cycle (controller clock <10 MHz).

Auto mode is almost identical to surveillance mode. Instead of triggering an interrupt and stopping the controller, the controller automatically updates its settings to the newly calculated optimal value and continues to run.

LVDS SAMPLE LOGIC

A simplified diagram of the AD973x LVDS data sampling engine is shown in Figure 76 and the timing diagram is shown in Figure 77.

The incoming LVDS data is latched by the data sampling signal (DSS), which is derived from DATACLK_IN. The LVDS controller delays DATACLK_IN to create the data sampling signal (DSS), which is adjusted to sample the LVDS data in the center of the valid data window. The skew between the DATACLK_IN and the LVDS data bits (DB<13:0>) must be minimal for proper operation. Therefore, it is recommended that the DATACLK_IN be generated in the same manner as the LVDS data bits (DB<13:0>) with the same driver and data lines (that is, it should just be another LVDS data bit running a constant 01010101… sequence, as shown in Figure 94).

If the DATACLK_IN signal is stopped, the DACCLK continues to generate an output signal based on the last two values clocked into the registers that drive D1 and D2, as shown in Figure 76. If these two registers are not equal, a large output at a frequency of one-half f_{DAC} may be generated at the DAC output.

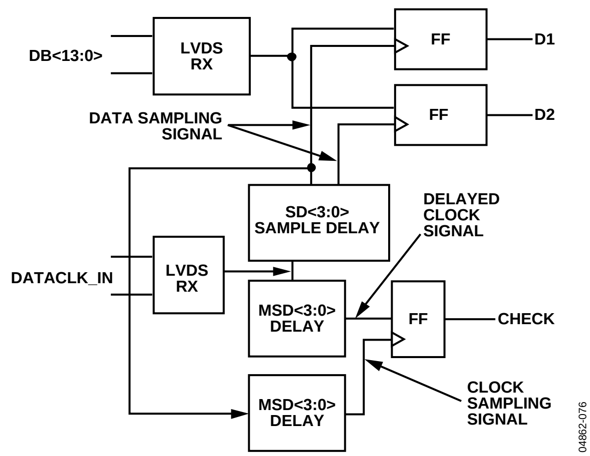


Figure 76. AD973x Internal LVDS Data Sampling Logic

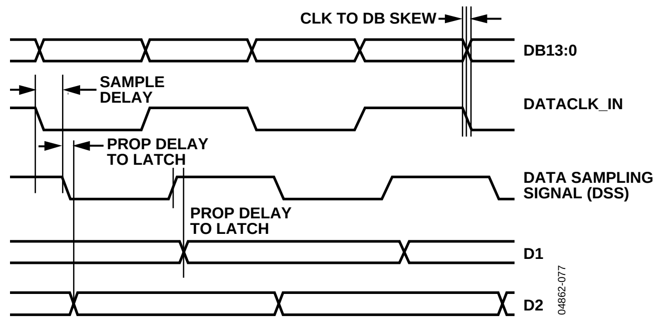


Figure 77. AD973x Internal LVDS Data Sampling Logic Timing

LVDS SAMPLE LOGIC CALIBRATION

The internal DSS delay must be calibrated to optimize the data sample timing. Once calibrated, the AD973x can generate an IRQ or automatically correct its timing if temperature or voltage variations change the timing too much. This calibration is done by using the delayed clock sampling signal (CSS) to sample the delayed clock signal (DCS). The LVDS sampling logic can find the edges of the DATACLK_IN signal and from this measurement the center of the valid data window can be located.

The internal delay line that derives the delayed DSS from DATACLK_IN is controlled by SD3:0 (Reg 05, Bits 7:4), while the DCS is controlled by MSD3:0 (Reg 04, Bits 7:4), and the CSS is controlled by MHD3:0 (Reg 04, Bits 3:0).

DATACLK_IN transitions must be time aligned with the LVDS data (DB<13:0>) transitions. This allows the CSS, derived from the DATACLK_IN, to find the valid data window of DB<13:0> by locating the DATACLK_IN edges. The latching (rising) edge of CSS is initially placed using Bits SD<3:0> and can then be shifted to the left using MSD<3:0> and to the right using MHD<3:0>. When CSS samples the DCS and the result is 1, (which can be read back via the check bit at Reg 05, Bit 0) the sampling is occurring in the correct data cycle. To find the leading edge of the data cycle, increment the measured setup delay until the check bit goes low. In order to find the trailing

edge, increment the measured hold delay (MHD) until check goes low. Always set MHD = 0 when incrementing MSD and vice versa.

The incremental units of SD, MSD, and MHD are in units of real time, not fractions of a clock cycle. The nominal step size is 80 ps.

OPERATING THE LVDS CONTROLLER IN MANUAL MODE VIA THE SPI PORT

The manual operation of the LVDS controller allows the user to step through both the setup and hold delays to calculate the optimal sampling delay (that is, the center of the data eye).

With SD<3:0> and MHD<3:0> set to 0, increment the setup time delay (MSD<3:0>, Reg 04, Bits 7:4) until the check bit (Reg 05, Bit 0) goes low and record this value. This locates the leading DATACLK_IN (and data) transition, as shown in Figure 78.

With SD<3:0> and MSD<3:0> set to 0, increment the hold time delay (MHD<3:0>, Reg 04, Bits 3:0) until the check bit (Reg 05, Bit 0) goes low and record this value. This locates the trailing DATACLK_IN (and DB<13:0>) transition, as shown in Figure 79.

Once both DATACLK_IN edges are located, the sample delay (SD<3:0>, Reg 05, Bits 7:4) must be updated according to the following equation:

$$\textit{Sample Delay} = (\textit{MHD} - \textit{MSD})/2$$

After updating SD<3:0>, verify that the sampling signal is in the middle of the valid data window by adjusting both MHD and then MSD with the new sample delay until the check bit goes low. The new MHD and MSD values should be equal to or within one unit delay if SD<3:0> was set correctly.

MHD and MSD may not be equal to or within one unit delay if the external clock jitter and noise exceeds the internal delay resolution. Differences of 2, 3, or more are possible and may require more filtering to provide stable operation.

The sample delay calibration should be performed prior to enabling surveillance mode or auto mode.

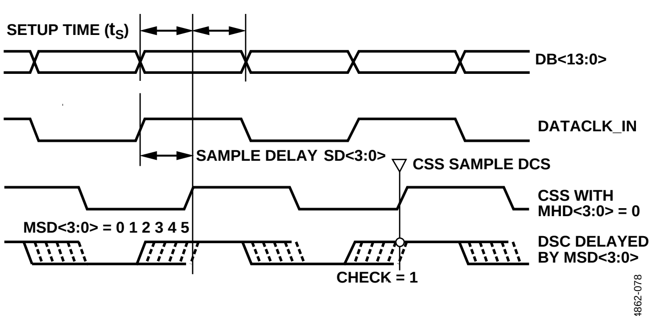


Figure 78. Setup Delay Measurement

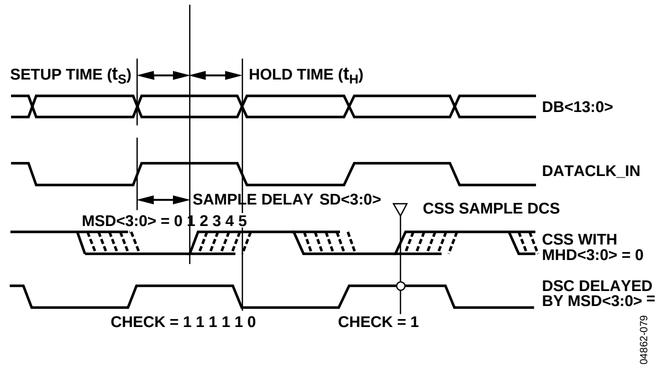


Figure 79. Hold Delay Measurement

OPERATING THE LVDS CONTROLLER IN SURVEILLANCE AND AUTO MODE

In surveillance mode, the controller searches for the edges of the data eye in the same manner as in the manual mode of operation and triggers an interrupt if the clock sampling signal (CSS) has moved more than the threshold value set by LTHR<1:0> (Reg 06, Bits 1:0).

There is an internal filter that averages the setup and hold time measurements to filter out noise and glitches on the clock lines.

$$\textit{Average Value} = (\textit{MHD} - \textit{MSD})/2$$
$$\textit{New Average} = \textit{Average Value} + (\Delta\ \textit{Average}/2\ \hat{}\ \textit{LFLT}<3:0>)$$

If an accumulating error in the average value causes it to exceed the threshold value (LTHR<1:0>), an interrupt is issued.

The maximum allowable value for LFLT<3:0> is 12. If LFLT<3:0> is too small, clock jitter and noise can cause erratic behavior. In most cases, LFLT can be set to the maximum value.

In surveillance mode, the ideal sampling point should first be found using manual mode and applied to the sample delay registers. The user should then set the threshold and filter values depending on how far the CSS signal is allowed to drift before an interrupt occurs. Then set the surveillance bit high (Reg 06, Bit 7) and monitor the interrupt signal either via the SPI port read back (Reg 01, Bit 7) or the IRQ pin.

In auto mode, the same steps should be taken to set up the sample delay, threshold, and filter length. To run the controller in auto mode, both the LAUTO (Reg 06, Bit 6) and LSURV (Reg 06, Bit 7) bits need to be set to 1. In auto mode, the LVDS interrupt should be set low (Reg 01, Bit 3) to allow the sample delay to be automatically updated if the threshold value is exceeded.

SYNC LOGIC AND CONTROLLER

A FIFO structure is utilized to synchronize the data transfer between the DACCLK and the DATACLK_IN clock domains. The sync controller writes data from DB<13:0> into an 8-word memory register based on a cyclic write counter clocked by the DSS, which is a delayed version of DACCLK_IN. The data is read out of the memory based on a second cyclic read counter clocked by DACCLK. The 8-word FIFO shown in Figure 80 provides sufficient margin to maintain proper timing under most conditions. The sync logic is designed to prevent the read and write pointers from crossing. If the timing drifts far enough to require an update of the phase offset (PHOF<1:0>), two samples are duplicated or dropped. Figure 81 shows the timing diagram for the sync logic.

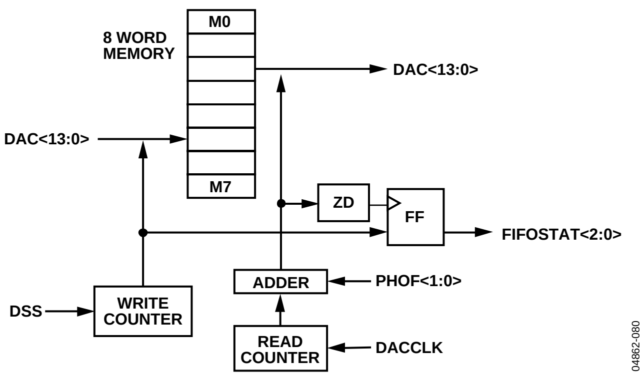


Figure 80. Sync Logic Block Diagram

SYNC LOGIC AND CONTROLLER OPERATION

The relationship between the readout pointer and the write pointer initially is unknown because the startup relationship between DACCLK and DATACLK_IN is unknown. The sync logic measures the relative phase between the two counters with the zero detect block and the flip-flop in Figure 80. The relative phase is returned in FIFOSTAT<2:0> (Reg 07, Bits 6:4), and sync logic errors are indicated by FIFOSTAT<3> (Reg 07, Bit 7). If FIFOSTAT<2:0> returns a value of 0 or 7, it signifies that the memory is sampling in a critical state (read and write pointers are close to crossing). If the FIFOSTAT<2:0> returns a value of 3 or 4, it signifies the memory is sampling at the optimal state (read and write pointers are farthest apart). If FIFOSTAT<2:0> returns a critical value, the pointer can be adjusted with the phase offset PHOF<1:0> (Reg 07, Bits 1:0). Due to the architecture of the FIFO, the phase offset can only adjust the read pointer in steps of 2.

OPERATING IN MANUAL MODE

To start operating the DAC in manual mode, allow DACCLK and DATACLK_IN to stabilize, then enable FIFO mode (Reg 00, Bit 2). Read FIFOSTAT<2:0> (Reg 07, Bits 6:4) to determine if adjustment is needed. For example, if FIFOSTAT<2:0> = 6, the timing is not yet critical but it is not optimal. To return to an optimal state (FIFOSTAT<2:0> = 4), the PHOF<1:0> (Reg 07, Bits 1:0) needs to be set to 1. Setting PHOF<1:0> = 1 effectively increments the read pointer by 2. This causes the write pointer value to be captured two clocks later, decreasing FIFOSTAT<2:0> from 6 to 4.

OPERATION IN SURVEILLANCE AND AUTO MODES

Once FIFOSTAT<2:0> has been manually placed in an optimal state, the AD973x sync logic can be run in surveillance or auto mode. To start, turn on surveillance mode by setting SSURV = 1 (Reg 08, Bit 7) then enable the sync interrupt (Reg 01, Bit 2). If STRH<0> = 0 (Reg 08, Bit 0), an interrupt occurs if FIFOSTAT<2:0> = 0 or 7. If STRH<0> = 1 (Reg 08, Bit 0), an interrupt occurs if FIFOSTAT<2:0> = 0, 1, 6, or 7. The interrupt can be read at Reg 01, Bit 6 at the AD973x IRQ pin.

To enter auto mode, complete the preceding steps then set SAUTO = 1 (Reg 08, Bit 6). Next, set the SYNC interrupt = 0 (Reg 01, Bit 2), to allow the phase offset (PHOF<1:0>) to be automatically updated if FIFOSTAT<2:0> violates the threshold value. The FIFOSTAT signal is filtered to improve noise immunity and reduce unnecessary phase offset updates. The filter operates with the following algorithm:

$$FIFOSTAT = FIFOSTAT + \Delta FIFOSTAT/2 \wedge SFLT<3:0>$$

where $0 \leq SFLT<3:0> \leq 12$. Values greater than 12 are set to 12. If SFLT<3:0> is too small, clock jitter and noise can cause erratic behavior. Normally SFLT can be set to the maximum value.

FIFO BYPASS

When the FIFO_MODE bit (Reg 01, Bit 2) is set to 0, the FIFO is bypassed with a mux. When the FIFO is enabled, the pipeline delay through the AD973x increases by the delta between the FIFO read pointer and write pointer plus 4 more clock periods.

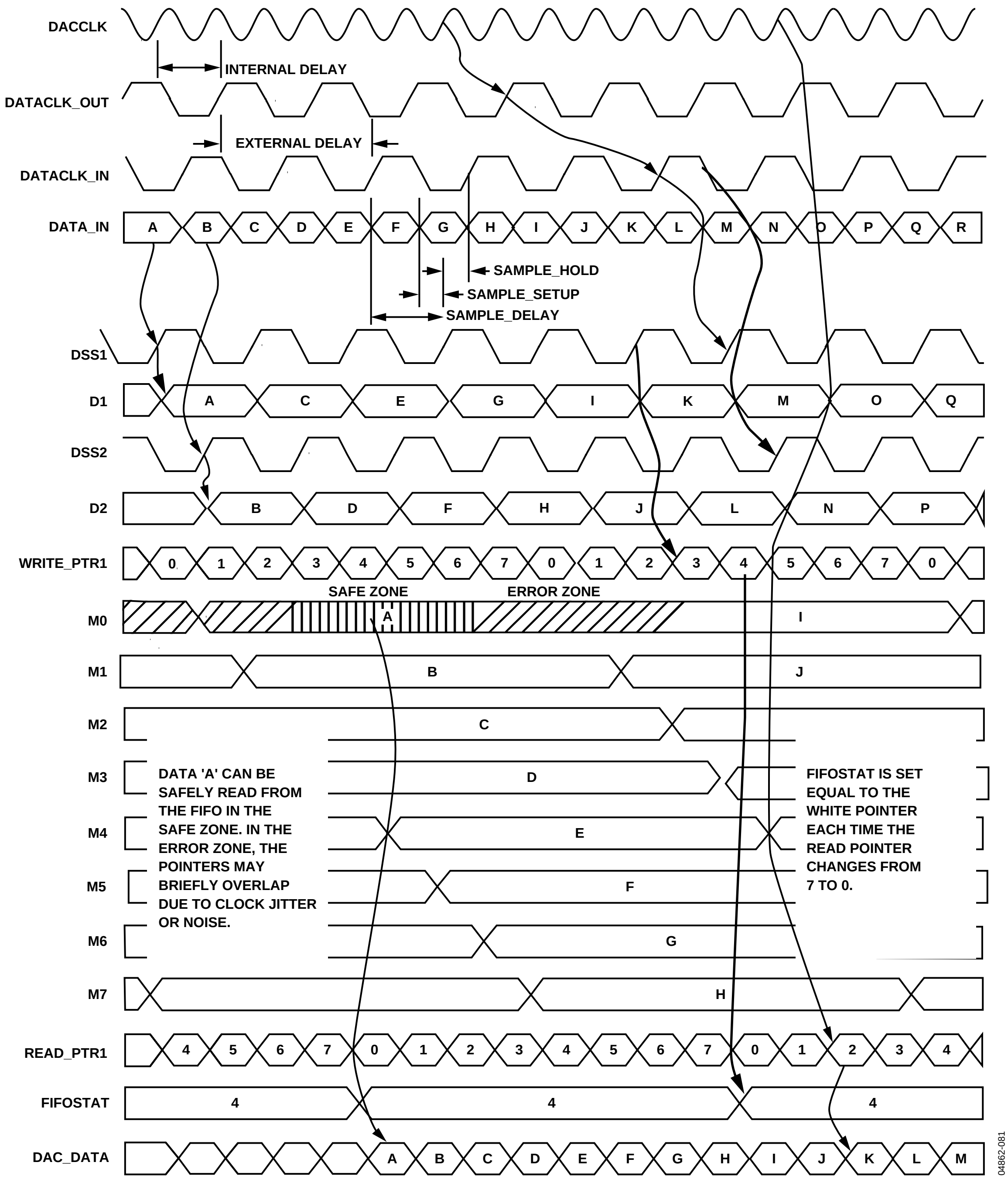


Figure 81. SYNC Logic Timing Diagram

DIGITAL BUILT-IN SELF TEST (BIST)

OVERVIEW

The AD973x includes an internal signature generator that processes incoming data to create unique signatures. These signatures can be read back from the SPI port, allowing verification of correct data transfer into the AD973x. BIST vectors provided on the AD973x-EB evaluation board CD can be used to check the full width data input or individual bits for PCB debug, utilizing the procedure in the AD973x BIST Procedure section. Alternatively, any vector may be used provided the expected signature is calculated in advance. The MATLAB® routine in the Generating Expected Signatures section may be used to calculate the expected signature. BIST should be used to verify correct data transfer because not all errors may be evident on a spectrum analyzer. There are four BIST signature generators that can be read back using Register 18 to Register 21, based on the setting of the BIST selection bits (Reg 17, Bits 7:6), as shown in Table 23. The BIST signature returned from the AD973x depends on the digital input during the test. Because the filters in the DAC have memory, it is important to put the correct idle value on the DATA inputs to flush the memory prior to reading the BIST signature.

Placing the idle value on the data inputs also allows the BIST to be set up while the DAC clock is running. The idle value should be all 0s in unsigned mode (0x0000) and all 0s except for the MSB in twos complement mode (0x2000).

The BIST consists of two stages; the first stage is after the LVDS receiver and the second stage is after the FIFO. The first BIST stage verifies correct sampling of the data from the LVDS bus while the second BIST stage verifies correct synchronization between the DAC_CLK domain and the DATACLK_IN domain. The BIST vector is generated using 32-bit LFSR signature logic. Because the internal architecture is a 2-bus parallel system, there are two 32-bit LFSR signature logic blocks on the both the LVDS and sync blocks. Figure 82 shows where the LVDS and sync phases are located.

Table 23. BIST Selection Bits

Bit	SEL<1>	SEL<0>
LVDS Phase 1	0	0
LVDS Phase 2	0	1
SYNC Phase 1	1	0
SYNC Phase 2	1	1

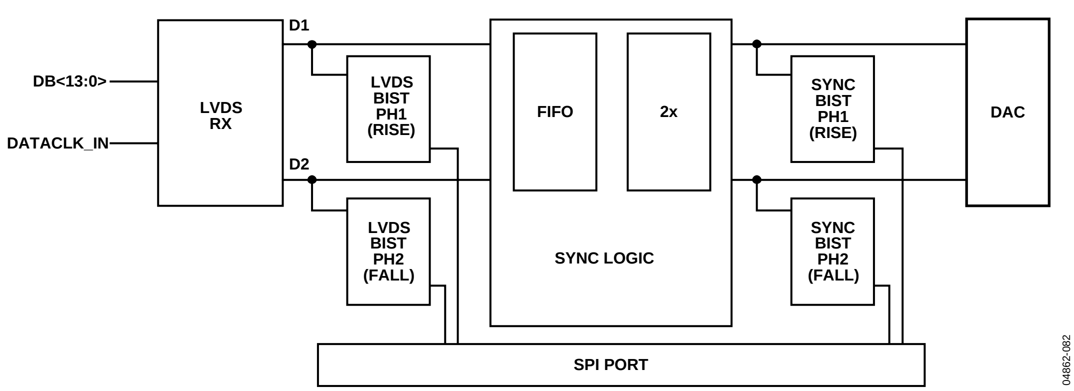


Figure 82. Block Diagram Showing LVDS and Sync Phase 1 and Phase 2

AD973x BIST PROCEDURE

1. Set RESET pin = 1.
2. Set input DATA = 0x0000 for signed (0x2000 for unsigned).
3. Enable DATACLK_IN if it is not already running.
4. Run for at least 16 DATACLK_IN cycles.
5. Set RESET pin = 0.
6. Run for at least 16 DATACLK_IN cycles.
7. Set RESET pin = 1.
8. Run for at least 16 DATACLK_IN cycles.
9. Set RESET pin = 0.
10. Set desired operating mode (1× mode and signed data are default values and expected for the supplied BIST vectors).
11. Set CLEAR (Reg 17, Bit 0), SYNC_EN (Reg 17, Bit 1) and LVDS_EN (Reg 17, Bit 2) high.
12. Wait 50 DATACLK_IN cycles to allow 0s to propagate through and clear sync signatures.
13. Set CLEAR low.
14. Read all signature registers (REG 21, 20, 19, and 18) for each of the four SEL (Reg 17, Bits 7:6) values and verify they are all 0x00.
    LVDS Phase 1
    a. Reg 17 set to 0x26 (SEL1 = 0, SEL0 = 0, SIG_READ = 1, LVDS_EN = 1, SYNC_EN = 1).
    b. Read Registers 20, 19, 18, and 17.
    LVDS Phase 2
    a. Reg 17 set to 0x66 (SEL1= 0, SEL0 = 1, SIG_READ = 1, LVDS_EN = 1, SYNC_EN = 1).
    b. Read Registers 20, 19, 18, and 17.
    SYNC Phase 1
    a. Reg 17 set to 0xA6 (SEL1= 1, SEL0 = 0, SIG_READ = 1, LVDS_EN = 1, SYNC_EN = 1).
    b. Read Registers 20, 19, 18, and 17.
    SYNC Phase 2
    a. Reg 17 set to 0xE6 (SEL1= 1, SEL0 = 1, SIG_READ = 1, LVDS_EN = 1, SYNC_EN = 1).
    b. Read Registers 20, 19, 18, and 17.
15. Clock the BIST vector into the AD973x.
16. After the BIST vector has been clocked into the part, hold DATA = 0x0000 for signed (0x2000 for unsigned); otherwise, the additional nonzero data changes the signature.
17. Read all signature registers (Reg 21, 20, 19, and 18 as described in Step 14) for each of the four SEL (Reg 17, Bits 7:6) values, and verify that they match the expected signatures shown in Table 24.
18. In some cases, the BIST circuitry may not be completely cleared, and an incorrect signature may be read. If this occurs, loop back to Step 11 and rerun the test to obtain the correct result. In an automated test, it may be best to always run the vector twice to assure a correct result.

AD973x EXPECTED BIST SIGNATURES

The BIST vectors provided on the AD973x-EB CD are in signed mode, so no programming needs to be done to the part to pass the BIST. The BIST vector is for 1×, no FIFO, and signed data.

For testing all 14 input bits, use the vector all_bits_unsnew.txt and verify against the signatures in Table 24.

Table 24. Expected BIST Data Readback for All Bits

LVDS Phase 1	LVDS Phase 2	SYNC Phase 1	SYNC Phase 2
CF71487C	66DF5250	CF71487C	66DF5250

For individual bit tests, use the vectors named bit*n*.txt (where *n* is the desired bit number being tested) and compare them against the values in Table 25.

Table 25. Expected BIST Data Readback for Individual Bits

Vector	Bit No.	LVDS Rise Expected	LVDS Fall Expected
bit0.txt	0	AABF0A00	2A400500
bit1.txt	1	2BBF0A00	6B400500
bit2.txt	2	29BE0A00	E9400500
bit3.txt	3	2DBC0A00	ED410500
bit4.txt	4	25B80A00	E5430500
bit5.txt	5	35B00A00	F5470500
bit6.txt	6	15A00A00	D54F0500
bit7.txt	7	55800A00	955F0500
bit8.txt	8	D5C00A00	157F0500
bit9.txt	9	D5410A00	153E0500
bit10.txt	10	D5430B00	15BC0500
bit11.txt	11	D5470900	15B80400
bit12.txt	12	D54F0D00	15B00600
bit13.txt	13	D55F0500	15A00200

Note the following for Table 25:

- The term *rise* refers to Phase 1 and *fall* refers to Phase 2.
- Byte order is Decimal Register Address 21, 20, 19, then 18.
- Sync phase should always equal LVDS phase in 1× mode.

GENERATING EXPECTED SIGNATURES

The following MATLAB code duplicates the internal logic of the AD973x. To use it, save this code in a file called bist.m.

```
--- begin bist.m ---
function [ ret1 , ret2] = bist(vec)
ret1 = bist1(vec(1:2:length(vec)-1));
ret2 = bist1(vec(2:2:length(vec)));
function ret = bist1(v)
sum = zeros(1,32);
for i = 1 :length(v)
if v(i) ~= 0
su(1) = ~xor(sum(32) ,bitget(v(i),1));
su(2) = ~xor(sum(1) ,bitget(v(i),2));
su(3) = ~xor(sum(2) ,bitget(v(i),3));
su(4) = ~xor(sum(3) ,bitget(v(i),4));
su(5) = ~xor(sum(4) ,bitget(v(i),5));
su(6) = ~xor(sum(5) ,bitget(v(i),6));
su(7) = ~xor(sum(6) ,bitget(v(i),7));
su(8) = ~xor(sum(7) ,bitget(v(i),8));
su(9) = ~xor(sum(8) ,bitget(v(i),9));
su(10) = ~xor(sum(9) ,bitget(v(i),10));
su(11) = ~xor(sum(10) ,bitget(v(i),11));
su(12) = ~xor(sum(11) ,bitget(v(i),12));
su(13) = ~xor(sum(12) ,bitget(v(i),13));
su(14) = ~xor(sum(13) ,bitget(v(i),14));
su(15) = sum(14); su(16) = sum(15);
su(17) = sum(16); su(18) = sum(17);
su(19) = sum(18); su(20) = sum(19);
su(21) = sum(20); su(22) = sum(21);
su(23) = sum(22); su(24) = sum(23);
su(25) = sum(24); su(26) = sum(25);
su(27) = sum(26); su(28) = sum(27);
su(29) = sum(28); su(30) = sum(29);
su(31) = sum(30); su(32) = sum(31);
sum = su;
end
end % for ret = dec2hex( 2.^[0:31]× sum',8);
--- end bist.m ---
```

To generate the expected BIST signatures, follow this procedure:

1. Start MATLAB and type the following at the command prompt:

   t = round(randn(1,100) × 2^{13}/8+2^{13}) ;
   [b1 b2] = bist(t)

   The first statement creates a random vector of 14-bit words, with a length of 100.

2. Set **t** equal to any desired vector, otherwise take this random vector and input it to the AD973x.

3. Alter the command **randn(1,100)** to change the vector length as desired.

4. Type **b1** at the command line to see the calculated signature for the LVDS BIST, Phase 1.

5. Type **b2** to see the value for LVDS BIST, Phase 2.

The values returned for **b1** and **b2** each are 32-bit hex values. They correspond to Reg 18, Reg 19, Reg 20, and Reg 21, where **b1** is the value read for SEL<1:0> = 0,0 (see Table 11) and **b2** is the value read for SEL<1:0> = 0,1.

When the DAC is in 1× mode, the signature at SYNC BIST, Phase 1 should equal the signature at LVDS BIST, Phase 1. The same is true for Phase 2.

CROSS CONTROLLER REGISTERS

The AD973x differential output stage can be adjusted in order to equalize the charge injection into the positive and negative outputs. This adjustment can impact certain performance characteristics, such as harmonic distortion or IMD. System performance can be enhanced by adjusting the cross controller as described next.

If the system is calibrated after manufacture, the cross controller offsets may be adjusted to provide optimum performance. Start by incrementing DNDEL<5:0> (Reg 11, Bits 5:0) while observing HD2 (second harmonic distortion) and/or IMD to find the desired optimum. If DNDEL does not influence the performance, set it to 0 and increment UPDEL<5:0> (Reg 10, Bits 5:0). Based on system characterization, it may be found that setting one or the other of these controls to the maximum value yields the best performance.

Figure 83 shows the effect of UPDEL and DNDEL.

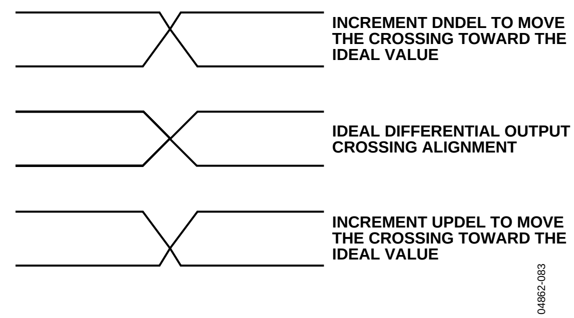


Figure 83. Effect of UPDEL and DNDEL

ANALOG CONTROL REGISTERS

The AD973x includes some registers for optimizing its analog performance. These registers include temperature trim for the band gap, noise reduction in the output current mirror, and output current mirror headroom adjustments.

BAND GAP TEMPERATURE CHARACTERISTIC TRIM BITS

Using TRMBG<2:0> (Reg 14, Bits 2:0) the temperature characteristic of the internal band gap can be trimmed to minimize the drift over temperature, as shown in Figure 84.

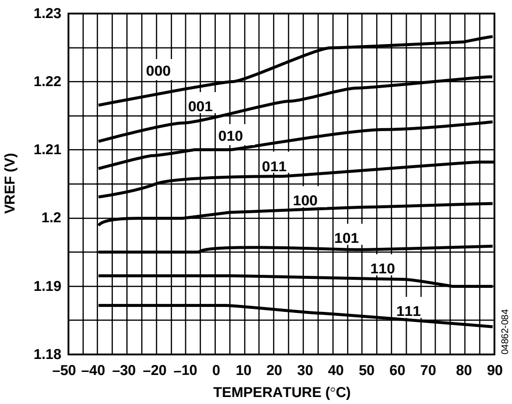


Figure 84. Band Gap Temperature Characteristic for Various TRMBG Values

The temperature changes are sensitive to process variations, and Figure 84 may not be representative of all fabrication lots. Optimum adjustment requires measurement of the device operation at two temperatures and development of a trim algorithm to program the correct TRMBG<2:0> values in external nonvolatile memory.

MIRROR ROLL-OFF FREQUENCY CONTROL

With MSEL<1:0> (Reg 14, Bits 7:6) the user can adjust the noise contribution of the internal current mirror to optimize the 1/f noise. Figure 85 shows MSEL vs. the 1/f noise with 20 mA full-scale current into a 50 Ω resistor.

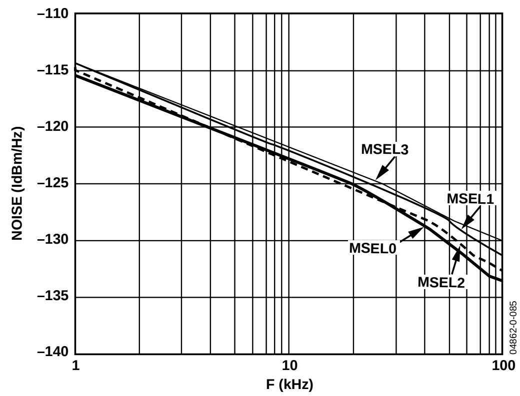


Figure 85. 1/f Noise with Respect to MSEL Bits

HEADROOM BITS

HDRM<7:0> (Reg 15, Bits 7:0) are for internal evaluation. It is not recommended to change the default reset values.

VOLTAGE REFERENCE

The AD973x output current is set by a combination of digital control bits and the I120 reference current, as shown in Figure 86.

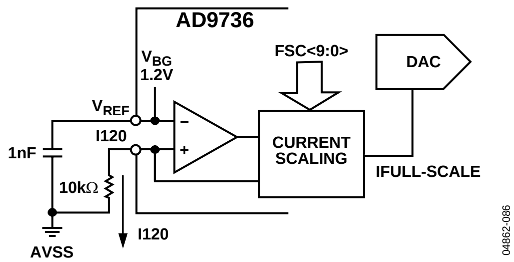


Figure 86. Voltage Reference Circuit

The reference current is obtained by forcing the band gap voltage across an external 10 kΩ resistor from I120 (Pin B14) to ground. The 1.2 V nominal band gap voltage (V_{REF}) generates a 120 μA reference current in the 10 kΩ resistor. This current is adjusted digitally by FSC<9:0> (Reg 02, Reg 03) to set the output full-scale current I_{FS}:

$$I_{FS} = \frac{V_{REF}}{R} \times \left(72 + \left(\frac{192}{1024} \times FSC < 9.0 >\right)\right)$$

The full-scale output current range is approximately 10 mA to 30 mA for register values from 0x000 to 0x3FF. The default value of 0x200 generates 20 mA full scale. The typical range is shown in Figure 87.

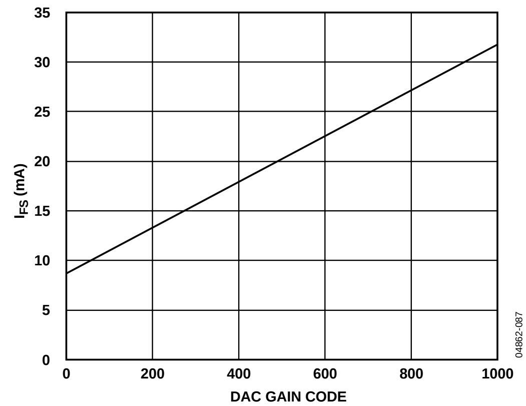


Figure 87. I_{FS} vs. DAC Gain Code

Always connect a 10 kΩ resistor from the I120 pin to ground and use the digital controls to vary the full-scale current. The AD973x is not a multiplying DAC. Applying an analog signal to I120 is not supported.

VREF (Pin C14) must be bypassed to ground with a 1 nF capacitor. The band gap voltage is present on this pin and may be buffered for use in external circuitry. The typical output impedance is near 5 kΩ. If desired, an external reference may be used to overdrive the internal reference by connecting it to the VREF pin.

IPTAT (Pin D14) is used for factory testing. It should be left floating.

APPLICATIONS INFORMATION

DRIVING THE DACCLK INPUT

The DACCLK input requires a low jitter differential drive signal. It is a PMOS input differential pair powered from the 1.8 V supply, so it is important to maintain the specified 400 mV input common-mode voltage. Each input pin can safely swing from 200 mV p-p to 800 mV p-p about the 400 mV common-mode voltage. While these input levels are not directly LVDS compatible, DACCLK may be driven by an offset ac-coupled LVDS signal, as shown in Figure 88.

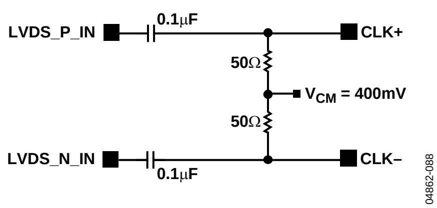


Figure 88. LVDS DACCLK Drive Circuit

If a clean sine clock is available, it may be transformer-coupled to DACCLK, as shown in Figure 105. Use of a CMOS or TTL clock may also be acceptable for lower sample rates. It can be routed through a CMOS to LVDS translator, then ac-coupled, as described previously. Alternatively, it may be transformer-coupled and clamped, as shown in Figure 89.

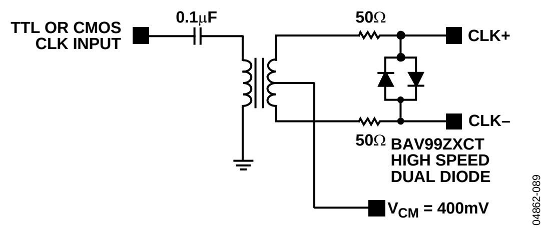


Figure 89. TTL or CMOS DACCLK Drive Circuit

A simple bias network for generating V_{CM} is shown in Figure 90. It is important to use CVDD18 and CVSS for the clock bias circuit. Any noise or other signal that is coupled onto the clock is multiplied by the DAC digital input signal and may degrade the DAC's performance.

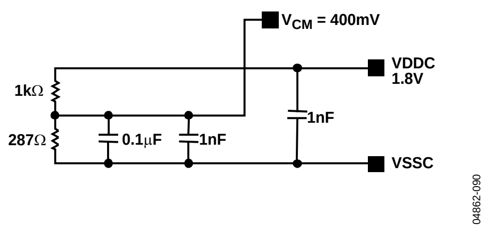


Figure 90. DACCLK V_{CM} Generator Circuit

DAC OUTPUT DISTORTION SOURCES

The second harmonic is mostly due to an imbalance in the output load. The dc transfer characteristic of the DAC is capable of second-harmonic distortion of at least −75 dBc. Output load imbalance or digital data noise coupling onto DACCLK causes additional second-harmonic distortion.

The DAC architecture inherently generates third harmonics, the levels of which depend on the output frequency and amplitude being generated. If any output signal is rectified and coupled back onto the DAC clock, it can generate additional third-harmonic energy.

The distortion components should be identical in amplitude and phase at both AD973x outputs. Even though each single-ended output includes a large amount of second-harmonic energy, a careful differential-to-single-ended conversion can remove most of it. Optimum performance at high intermediate frequency (IF) outputs is obtained with the output circuit shown in Figure 91. This is the configuration implemented on the evaluation board (Figure 105). The 20 Ω series resistors allow the DAC to drive a less reactive load, which improves distortion. Further improvement can be made by adding the balun T3 to help provide an equal load to both DAC outputs.

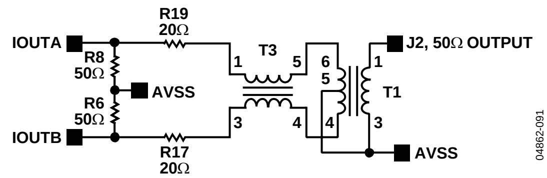


Figure 91. IF Signal Output Circuit

Because T1 has a differential input but a single-ended output, Pin 4 of T1 has a higher capacitance to ground due to parasitics to Pin 3. T1 Pin 6 has lower parasitic capacitance to ground because it drives 50 Ω at Pin 1. This presents an unbalanced load to the DAC output, so T3 is added to improve the load balancing. Refer to Figure 105 for the transformer part numbers.

DC-COUPLED DAC OUTPUTS

In some cases, it may be desirable to dc-couple the AD973x outputs. The best method for doing this is shown in Figure 92. This circuit can be used with voltage or current feedback amplifiers. Because the DAC output current is driving a virtual ground, this circuit may offer enhanced settling times. The settling time is limited by the op amp rather than the DAC. This circuit is intended for use where the amplifiers can be powered by a bipolar supply.

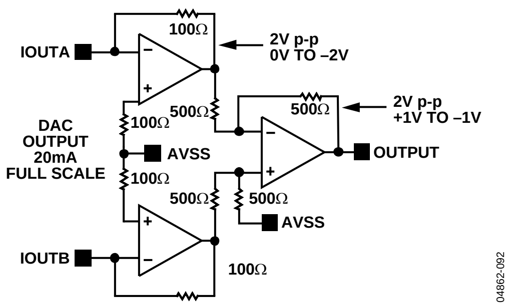


Figure 92. Op Amp I to V Conversion Output Circuit

An alternate circuit is shown in Figure 93. It suffers from dc offset at the output unless the DAC load resistors are small, relative to the amplifier gain and feedback resistors.

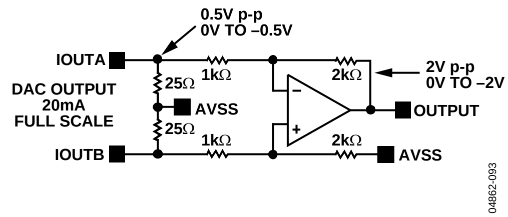


Figure 93. Differential Op Amp Output Circuit

DAC DATA SOURCES

The circuit shown in Figure 94 allows optimum data alignment when running the AD973x at full speed. This circuit can be easily implemented in the FPGA or ASIC used to drive the digital inputs. It is important to use the DATACLK_OUT signal because it helps to cancel some of the timing errors. In this configuration, DATACLK_OUT generates the DDR LVDS DATACLK_IN to drive the AD973x. The circuit aligns the DATACLK_IN and the digital input data (DB<13:0>) as required by the AD973x. The LVDS controller in the AD973x uses DATACLK_IN to generate the internal DSS to capture the incoming data in the center of the valid data window.

To operate in 2× mode, the circuit in Figure 94 must be modified to include a divide-by-2 block in the path of DATACLK_OUT. Without this additional divider, the data and DATACLK_IN runs 2× too fast. DATACLK_OUT is always DACCLK/2.

Contact FPGA vendors directly regarding the maximum output data rates supported by their products.

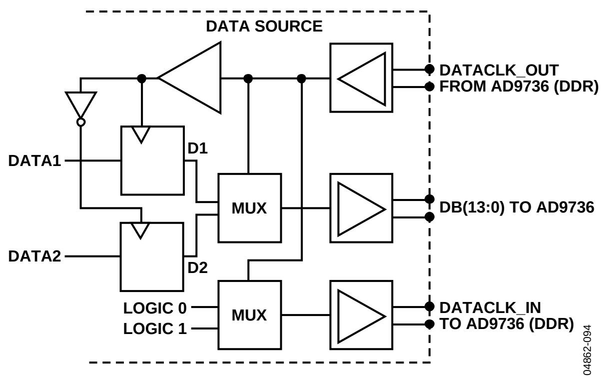


Figure 94. Recommended FPGA/ASIC Configuration for Driving AD973x Digital Inputs, 1× Mode

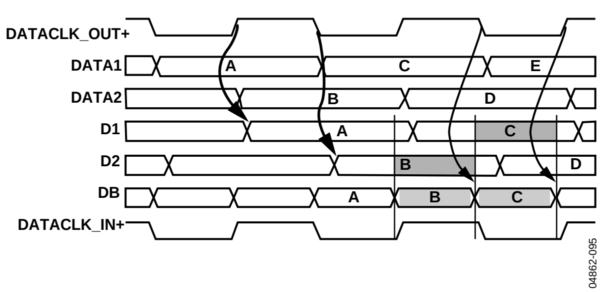


Figure 95. FPGA/ASIC Timing for Driving AD973x Digital Inputs, 1× Mode

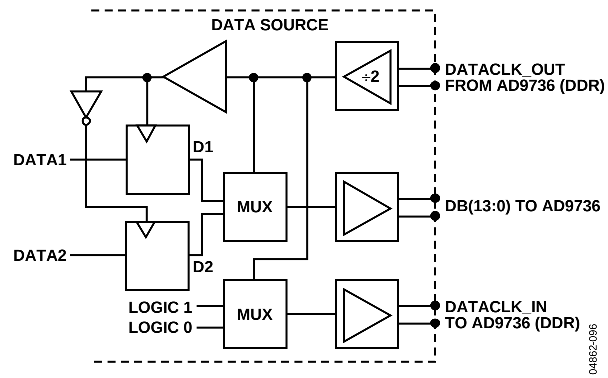


Figure 96. Recommended FPGA/ASIC Configuration for Driving AD973x Digital Inputs, 2× Mode

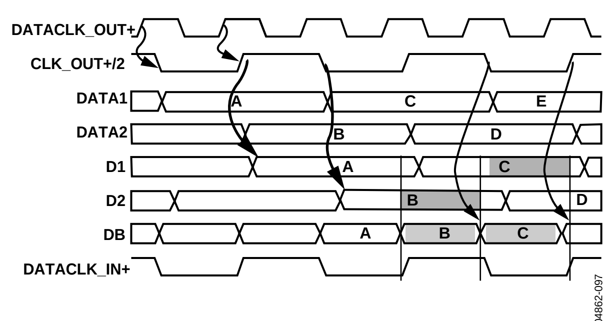


Figure 97. FPGA/ASIC Timing for Driving AD973x Digital Inputs, 2× Mode

INPUT DATA TIMING

The AD973x is intended to operate with the LVDS and sync controllers running to compensate for timing drift due to voltage and temperature variations. In this mode, the key to correct data capture is to present valid data for a minimum amount of time. The AD973x minimum valid data time was measured by increasing the input data rate to the point of failure. The nominal supply voltages were used and the temperature was set to the worst case of 85°C. The input data was verified via the BIST signature registers, because the DAC output does not run as fast as the input data logic. The following example explains how the minimum data valid period is calculated for the typical performance case.

These factors must be considered in determining the minimum valid data window at the receiver input:

- Data rise and fall times: 100 ps (rise + fall)
- Internal clock jitter: 10 ps (DATACLK_OUT + DATACLK_IN)
- Bit-to-bit skew: 50 ps
- Bit-to-DATACLK_IN skew: 50 ps
- Internal data sampling signal resolution: 80 ps

For nominal silicon, the BIST typically indicates failure at 2.15 GSPS or a DACCLK period of 465 ps. The valid data window is calculated by subtracting all the other variables from the total data period:

Minimum Data Valid Time = DACCLK Period − Data Rise − Data Fall − Jitter − Bit-to-Bit Skew − Bit-to-DATACLK_IN Skew − Data Sampling Signal Resolution

For the 400 mV p-p LVDS signal case:

Minimum Data Valid = 465 ps − 100 ps − 10 ps − 50 ps − 80 ps = 465 ps − 240 ps = 225 ps

For correct data capture, the input data must be valid for 225 ps. Slower edges, more jitter, or more skew require an increase in the clock period to maintain the minimum data valid period. Table 26 shows the typical minimum data valid period (t_{MDE}) for 400 mV p-p differential and 250 mV p-p differential LVDS swings.

The ability of the AD973x to capture incoming data is dependent on the speed of the silicon, which varies from lot to lot. The typical (or average) silicon speed operates with data that is valid for 225 ps at 85°C. Statistically, the worst extreme for slow silicon may require up to a 344 ps valid data period, as specified in Table 2.

Table 26. Typical Minimum Data Valid Times

Differential Input Voltage	BIST Max f_{CLK}	Min Clock Period	Typ Min Data Valid at Receiver
400 mV	2.15 GHz	465 ps	225 ps
250 mV	2.00 GHz	500 ps	260 ps

At 1.2 GHz, the typical 400 mV p-p minimum data valid period of 225 ps leaves 608 ps for external factors. Under the same conditions, the worst expected minimum data valid period of 344 ps leaves 489 ps for external data uncertainty.

The 100 mV LVDS V_{od} threshold test is a dc test to verify that the input logic state changes. It does not indicate the operating speed. The receiver's ability to recover the data depends on the input signal overdrive. With a 250 mV input, there is a 150 mV overdrive, and with a 400 mV signal, there is a 300 mV overdrive. The relationship between overdrive level and timing is very nonlinear. Higher levels of overdrive result in smaller minimum valid data windows.

For typical silicon, decreasing the LVDS swing from 400 mV p-p to 250 mV p-p requires the minimum data valid period to increase by 15%. This is illustrated in Figure 98.

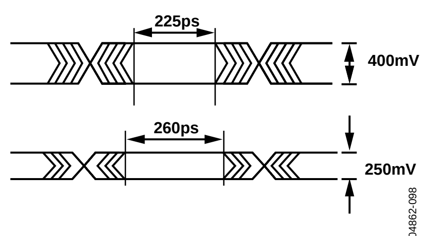


Figure 98. Typical Minimum Valid Data Time (t_{MDE}) vs. LVDS Swing

The minimum valid data window changes with temperature, voltage, and process. The maximum value presented in the specification table was determined from a 6σ distribution in the worst-case conditions.

SYNCHRONIZATION TIMING

When more than one AD973x must be synchronized or when a constant group delay must be maintained, the internal controllers cannot be used. If the FIFO is enabled, the delay between multiple AD973x devices is unknown. If the DATACLK_OUT from multiple devices is used, there is an uncertainty of two DACCLK periods because the initial phase of DATACLK_OUT with respect to DACCLK cannot be controlled. This means one DAC must be used to provide DATACLK_OUT for all synchronized DACs and all timing must be externally managed. The following timing information allows system timing to be calculated so that multiple AD973xs can be synchronized.

DATACLK_OUT changes relative to the rising edge of DACCLK+ and is delayed, as shown in Figure 99. Because DACCLK is divided by 2 to create DATACLK_OUT, the phase of DATACLK_OUT can be 0° or 180°. There is no way to predict or control this relationship. It may be different after each power cycle and is not affected by hardware or software resets.

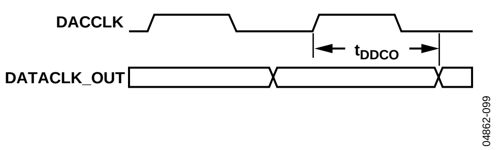


Figure 99. DACCLK to DATACLK_OUT Delay

The incoming data is de-interleaved internally as shown in Figure 76. Each edge of DATACLK_IN latches an incoming sample in two alternating registers. The DATACLK_IN to data setup and hold definitions are illustrated in Figure 100. All the data inputs must be valid during the setup-and-hold period. External skew effectively increases the setup and hold times that the data source must meet.

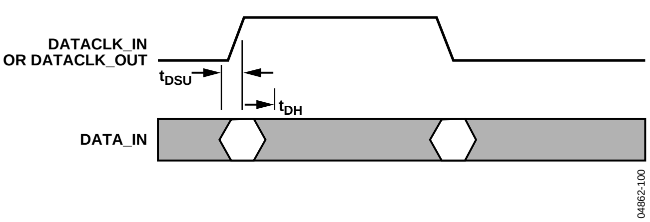


Figure 100. Standard Definitions for DATACLK_IN or DATACLK_OUT to Data Setup and Hold, SD = 0

While correct DATA_IN vs. DATACLK_IN timing is critical, the transition of the incoming data to the DACCLK domain is equally critical. By referencing the incoming DATA and DATACLK_IN timing to the DATACLK_OUT signal, some timing uncertainty can be removed. The DATACLK_OUT timing very closely tracks the timing of the DACCLK-controlled registers. Any variation in the path delay affects both paths in almost the same way. If DATACLK_OUT is not used, the full DACCLK to DATACLK_OUT path variation reduces the external timing margin. Figure 101 shows a simplified view of the internal clocking scheme with the relevant delay paths.

The internal architecture is interleaved such that each phase has twice as long to make the transition across the clock domains. This results in an extremely narrow window where the incoming data must be held stable.

Table 27 shows the timing parameters for Figure 99 and Figure 100. These parameters were measured for a sample of five devices from five silicon lots. Worst-case fast and slow skew lots were included in addition to the nominal (or average) lot. The typical −40°C to typical +85°C spread illustrates the variability with temperature for a single lot. Adding in lot-to-lot variation with the fast and slow lots indicates the worst-case spread in timing.

The timing varies such that all of the parameters move in the same direction. For example, if the DATACLK_IN to data setup time is fast, the hold time is similarly fast. The DACCLK to DATACLK_OUT delay and the DATACLK_OUT to data setup and hold is also at the fast end of the range.

Note that the polarities of setup-and-hold values in Table 27 conform to the standard convention of setup time occurring prior to the latching edge and hold time occurring after the latching edge, as shown in Figure 100.

Table 27. AD973x Clock and Data Timing Parameters

Symbol and Definition	Fast −40°C	Typ −40°C	All +25°C	Typ +85°C	Slow +85°C	Unit
t_{DDCO} – DACCLK to DATACLK_OUT Delay	1650	1800	1890	2050	2350	ps
t_{DCISU} – DATACLK_IN to DATA Setup	−100	−120	−150	−170	−220	ps
t_{DCIH} – DATACLK_IN to DATA Hold	210	220	240	280	360	ps
t_{DISU} – DATACLK_OUT to DATA Setup	1310	1440	1611	1710	1970	ps
t_{DIH} – DATACLK_OUT to DATA Hold	−1250	−1360	−1548	−1640	−1890	ps

POWER SUPPLY SEQUENCING

The 1.8 V supplies should be enabled simultaneously with or prior to the 3.3 V supplies. Do not enable the 3.3 V supplies when the 1.8 V supplies are off.

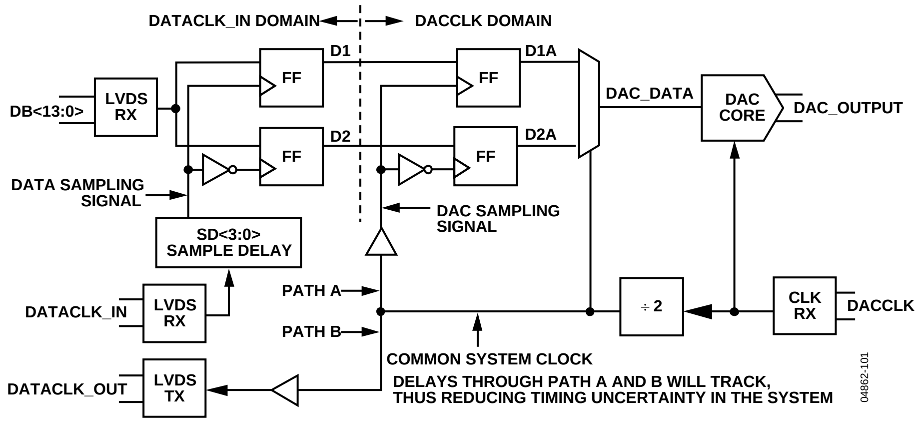


Figure 101. Simplified Internal Clock Routing

AD973x EVALUATION BOARD SCHEMATICS

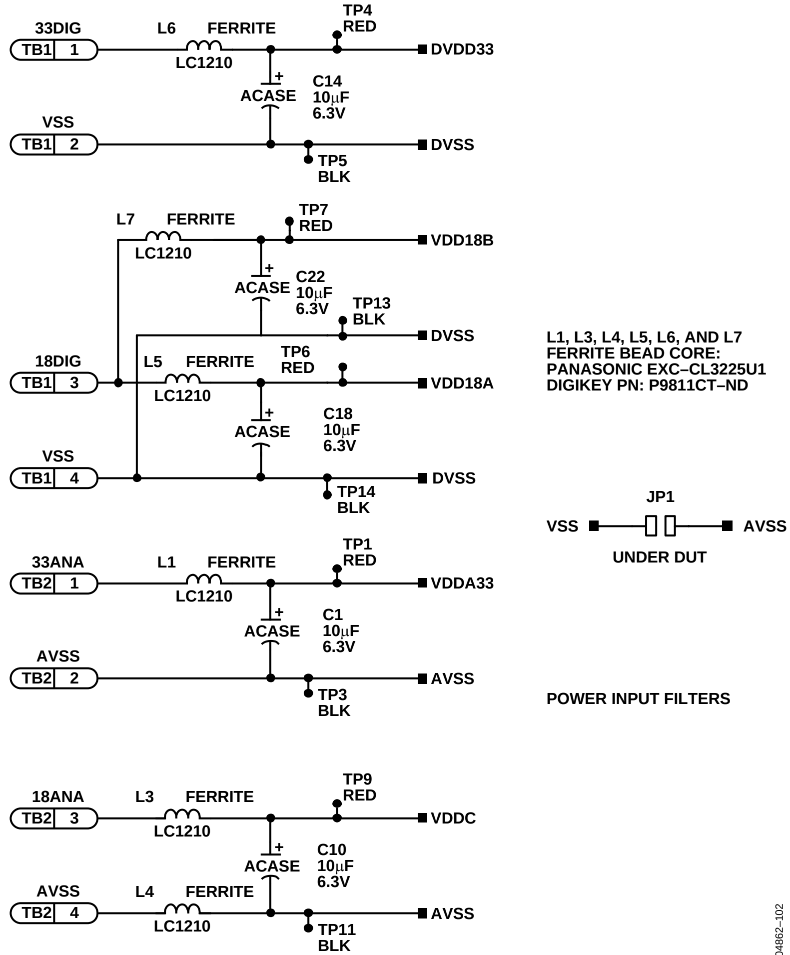


Figure 102. Power Supply Inputs for AD973x Evaluation Board, Rev. F

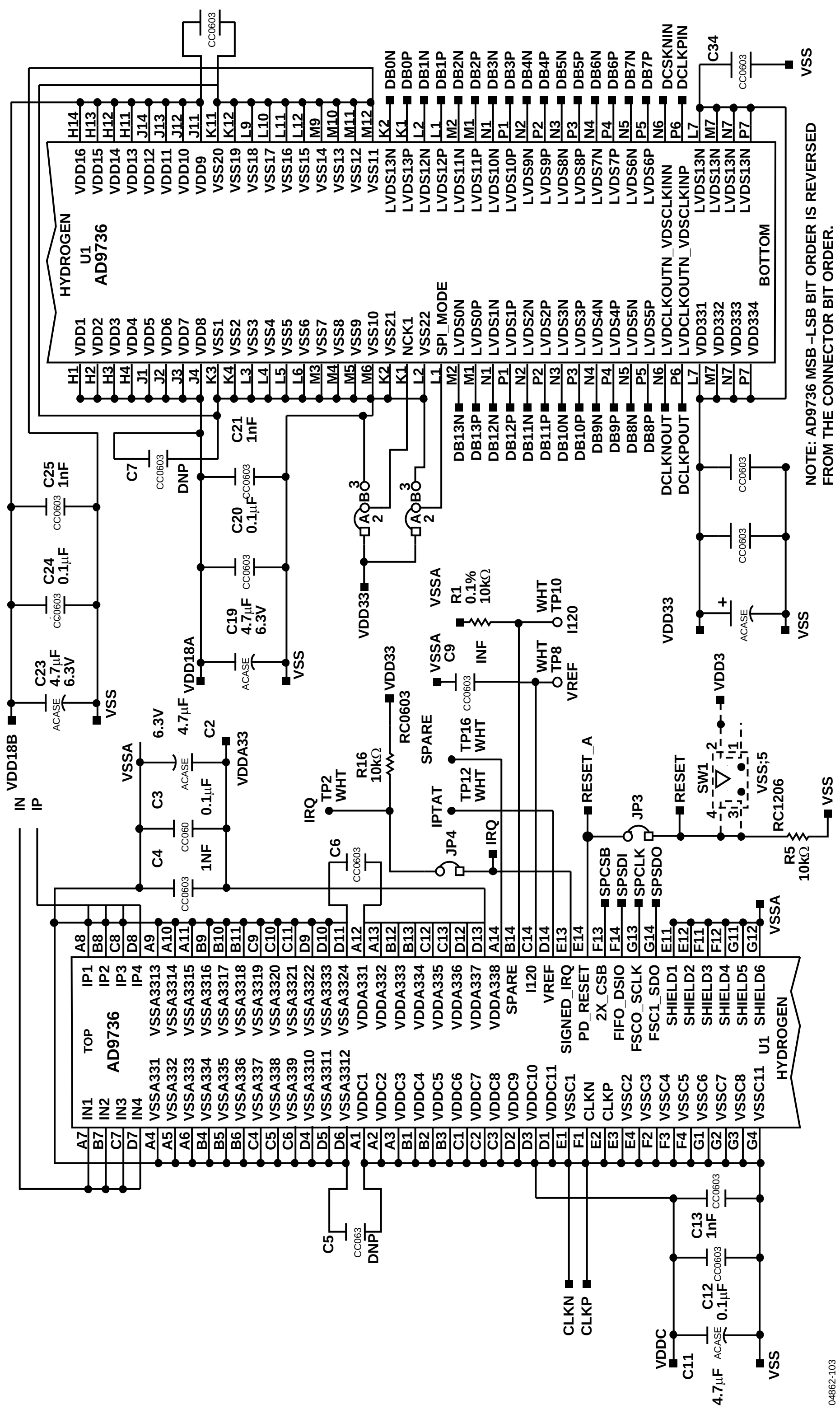


Figure 103. Circuitry Local to AD973x, Evaluation Board, Rev. F

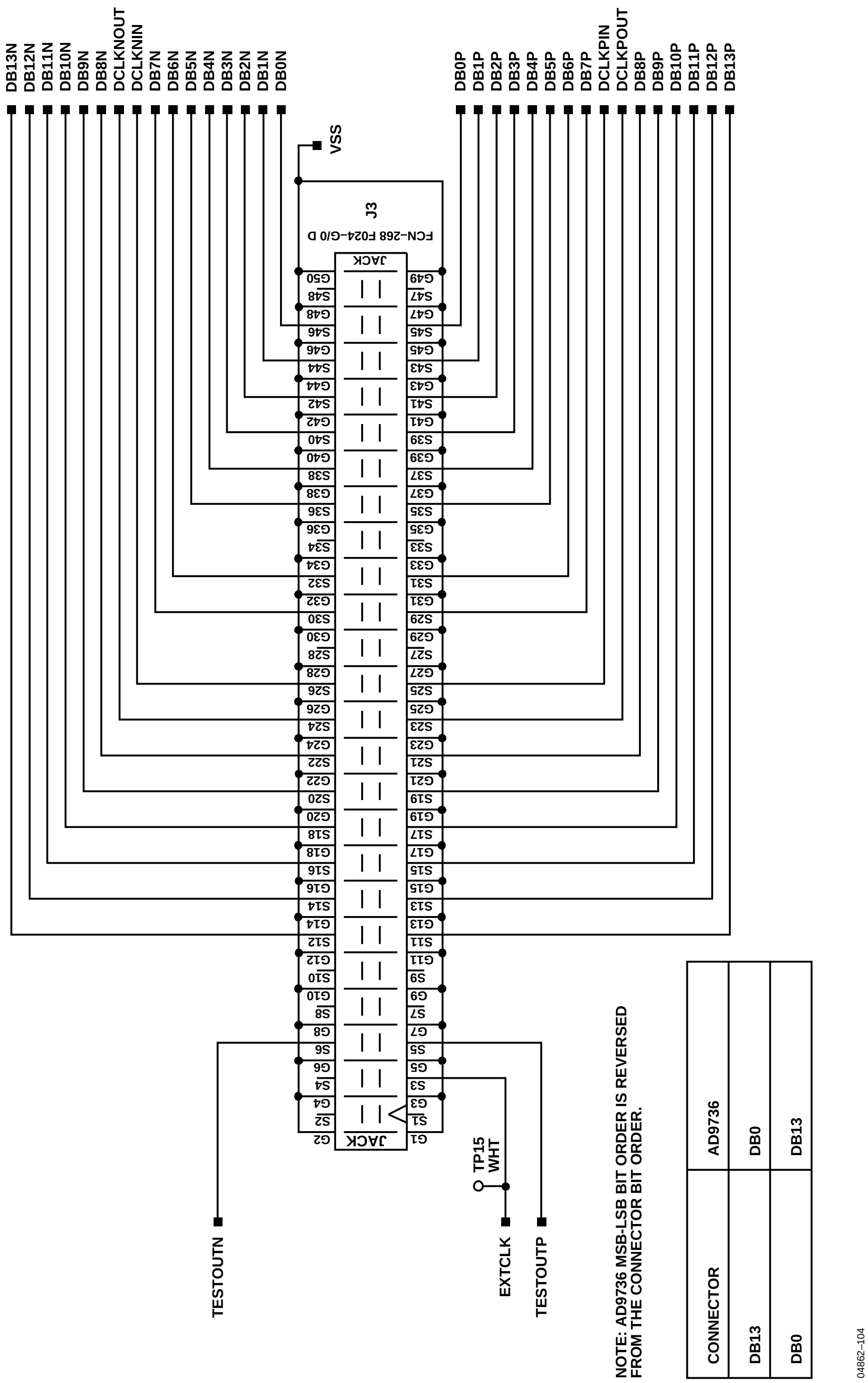


Figure 104. High Speed Digital I/O Connector, AD973x Evaluation Board, Rev. F

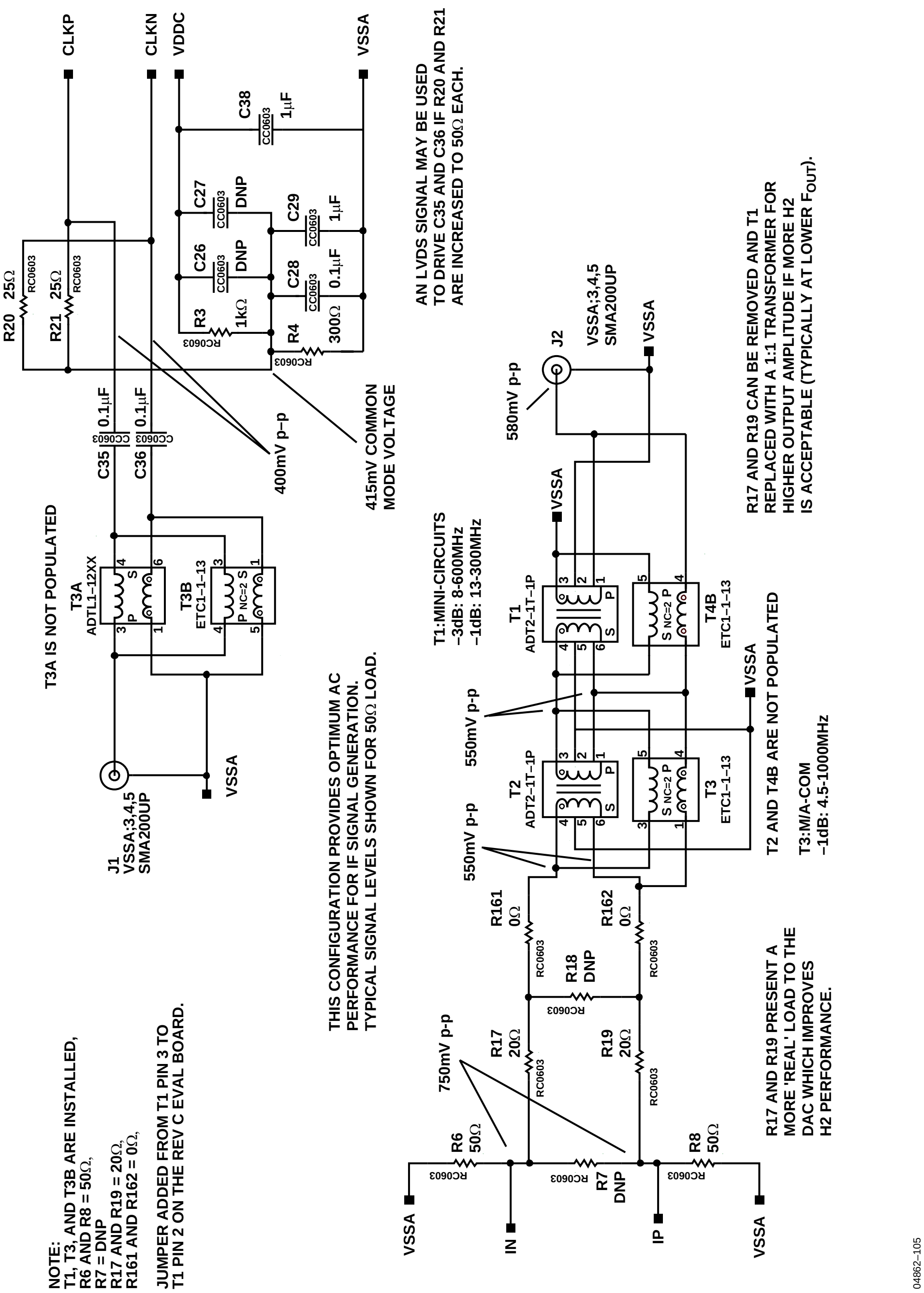


Figure 105. Clock Input and Analog Output, AD973x Evaluation Board, Rev. F

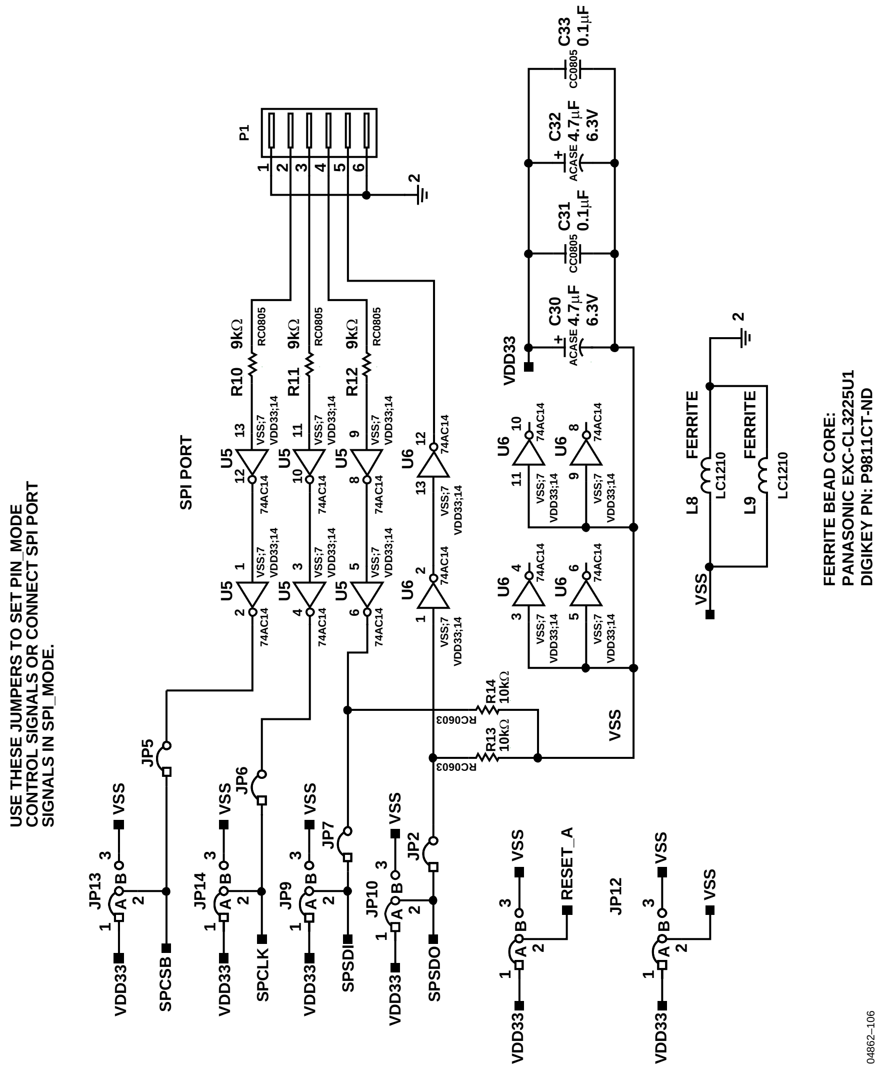


Figure 106. SPI Port Interface, AD973x Evaluation Board, Rev. F

AD973x EVALUATION BOARD PCB LAYOUT

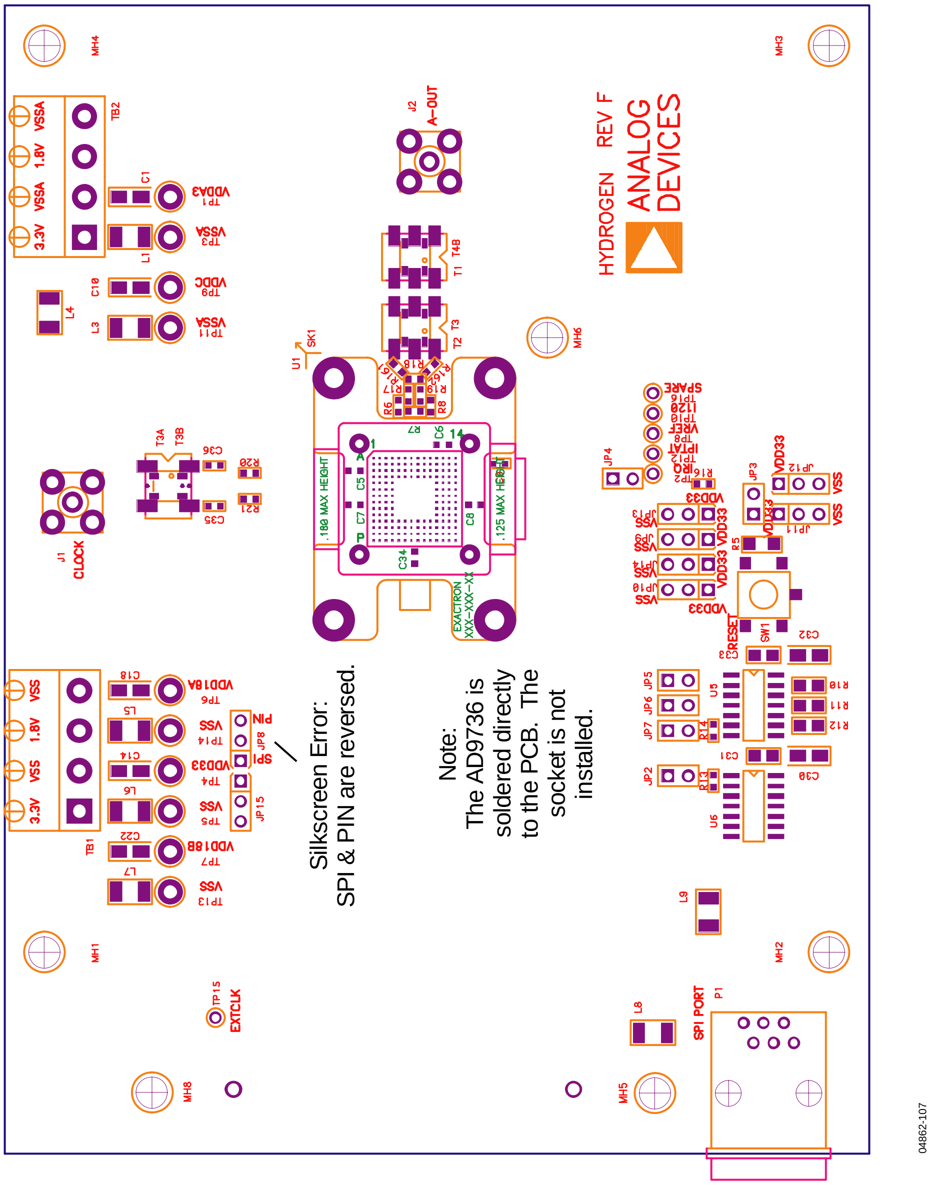


Figure 107. CB Layout Top Placement, AD973x Evaluation Board, Rev. F

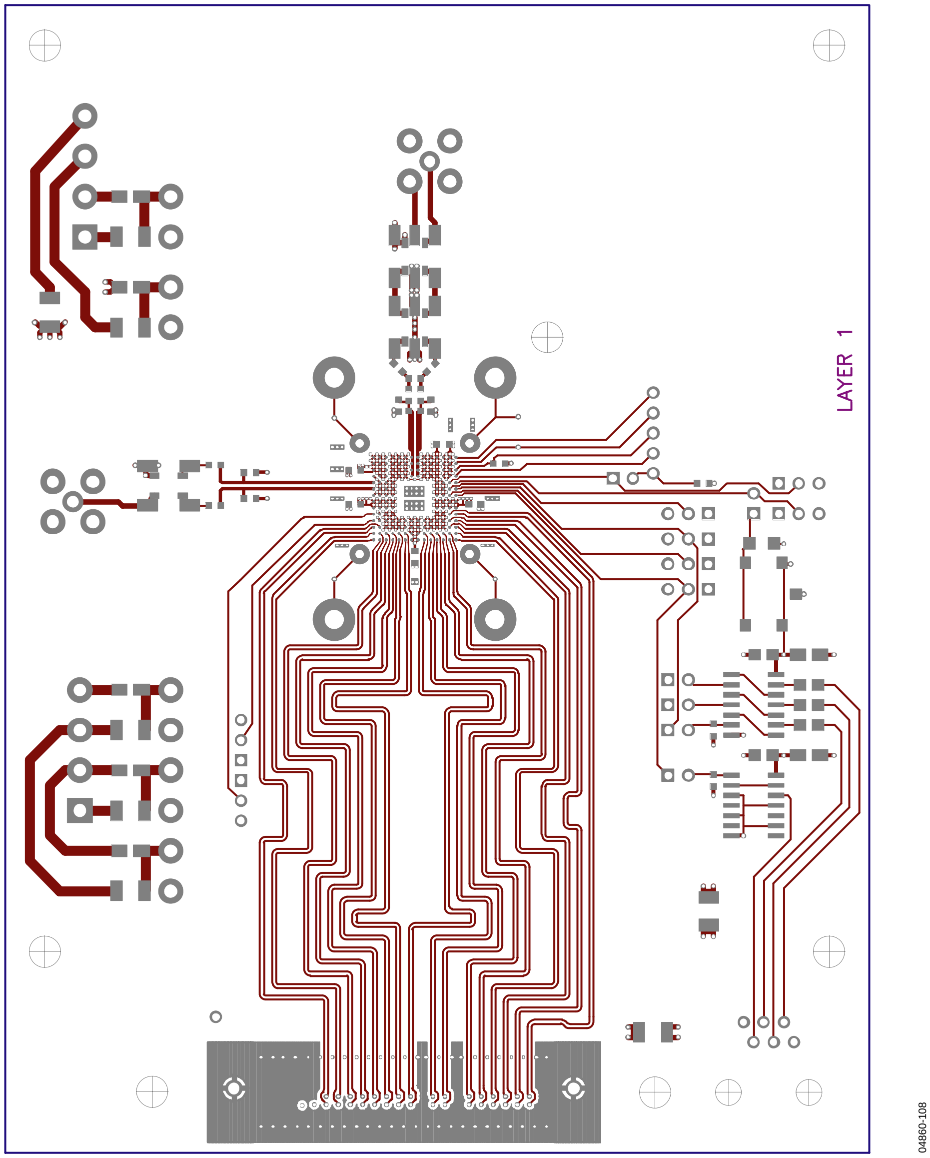


Figure 108. PCB Layout Layer 1, AD973x Evaluation Board, Rev. F

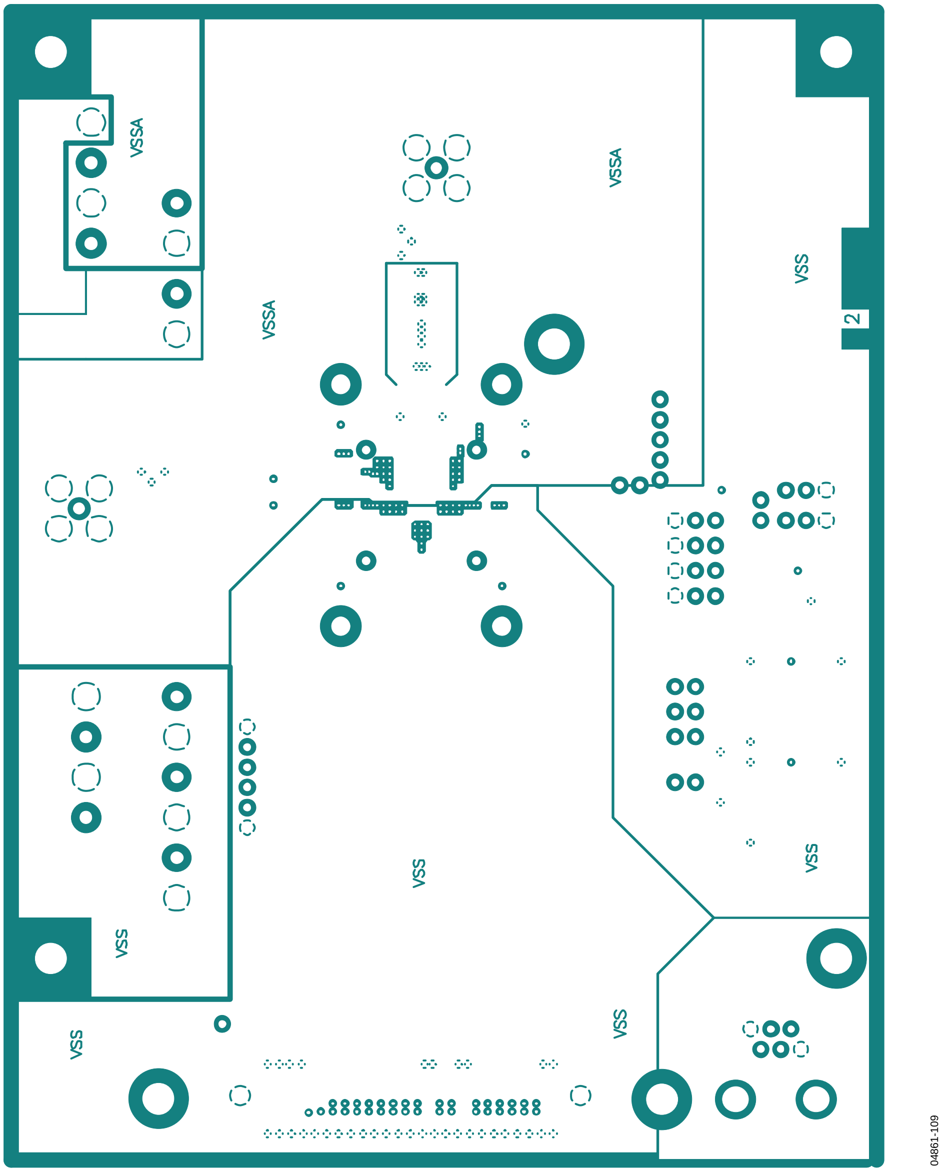


Figure 109. PCB Layout Layer 2, AD973x Evaluation Board, Rev. F

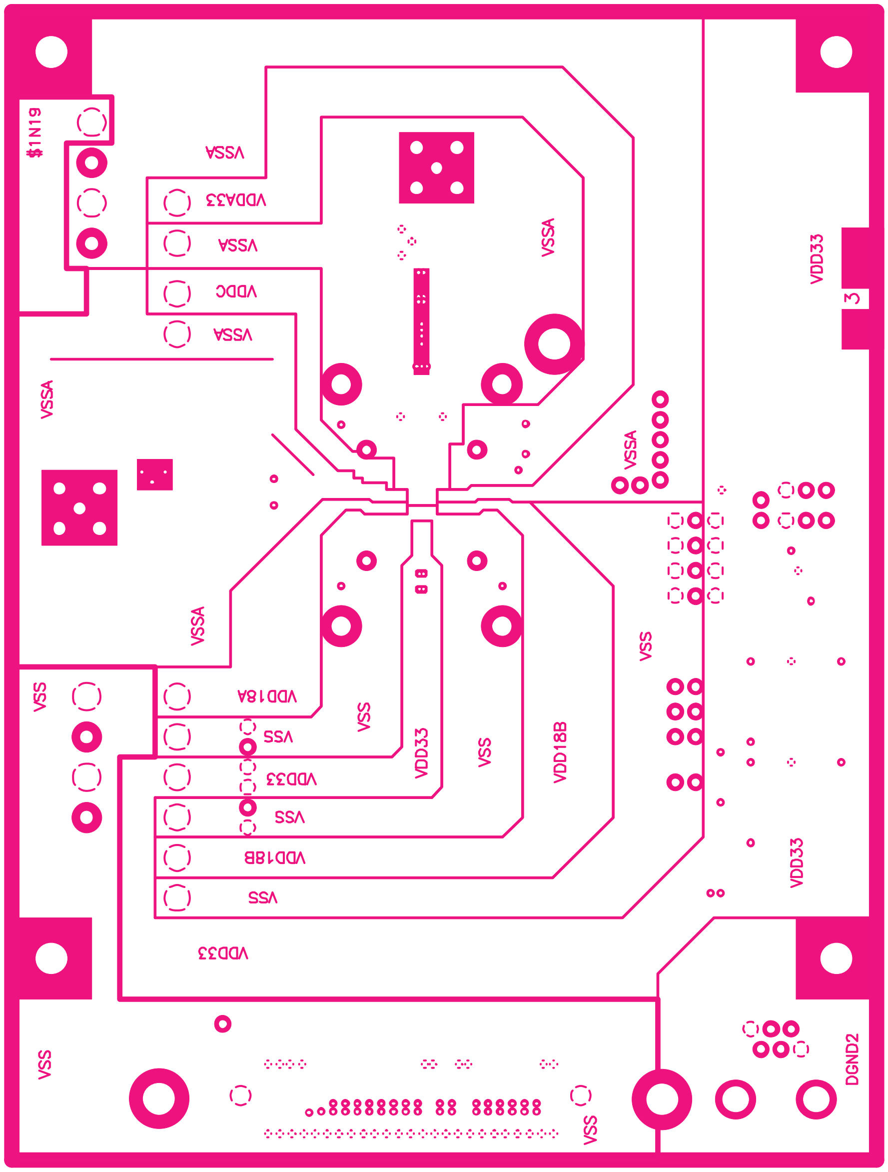


Figure 110. PCB Layout Layer 3, AD973x Evaluation Board, Rev. F

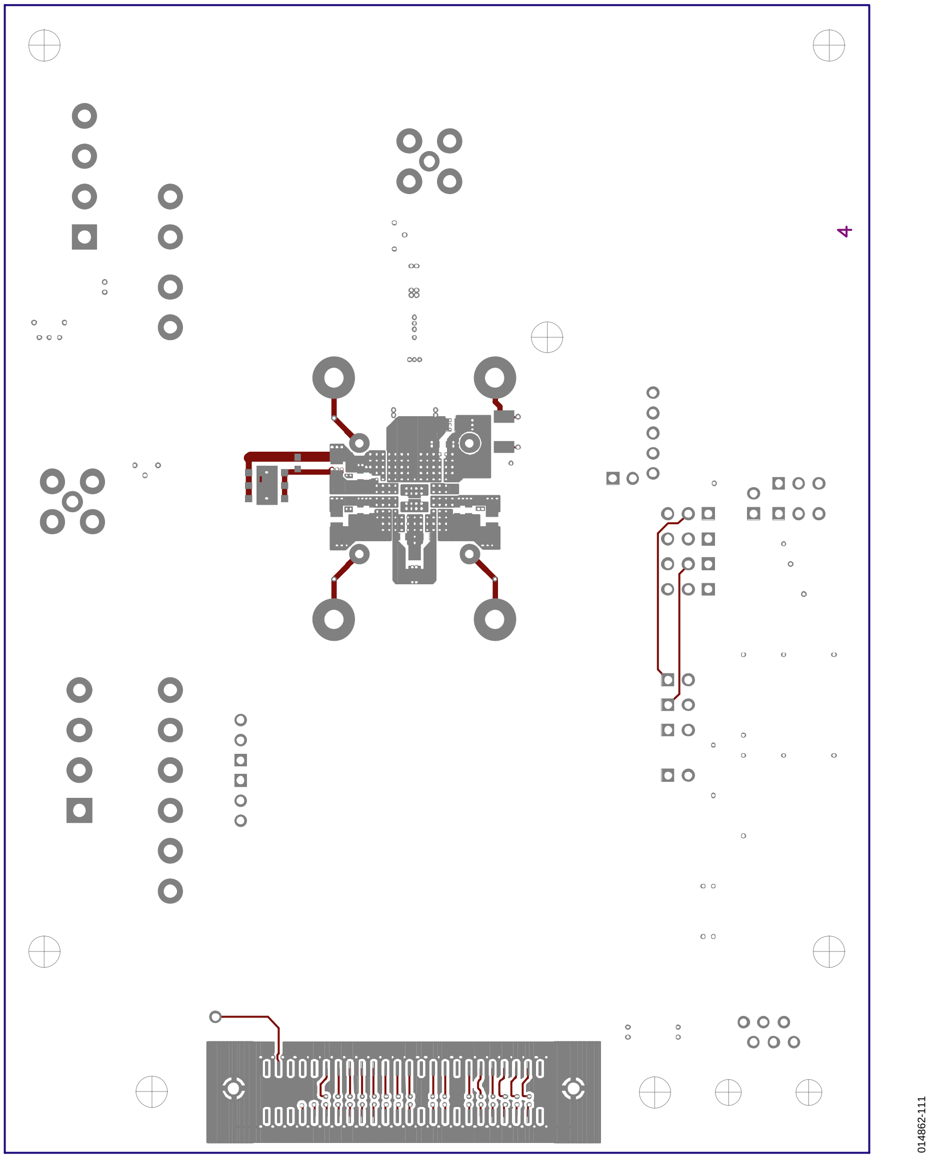


Figure 111. PCB Layout Layer 4, AD973x Evaluation Board, Rev. F

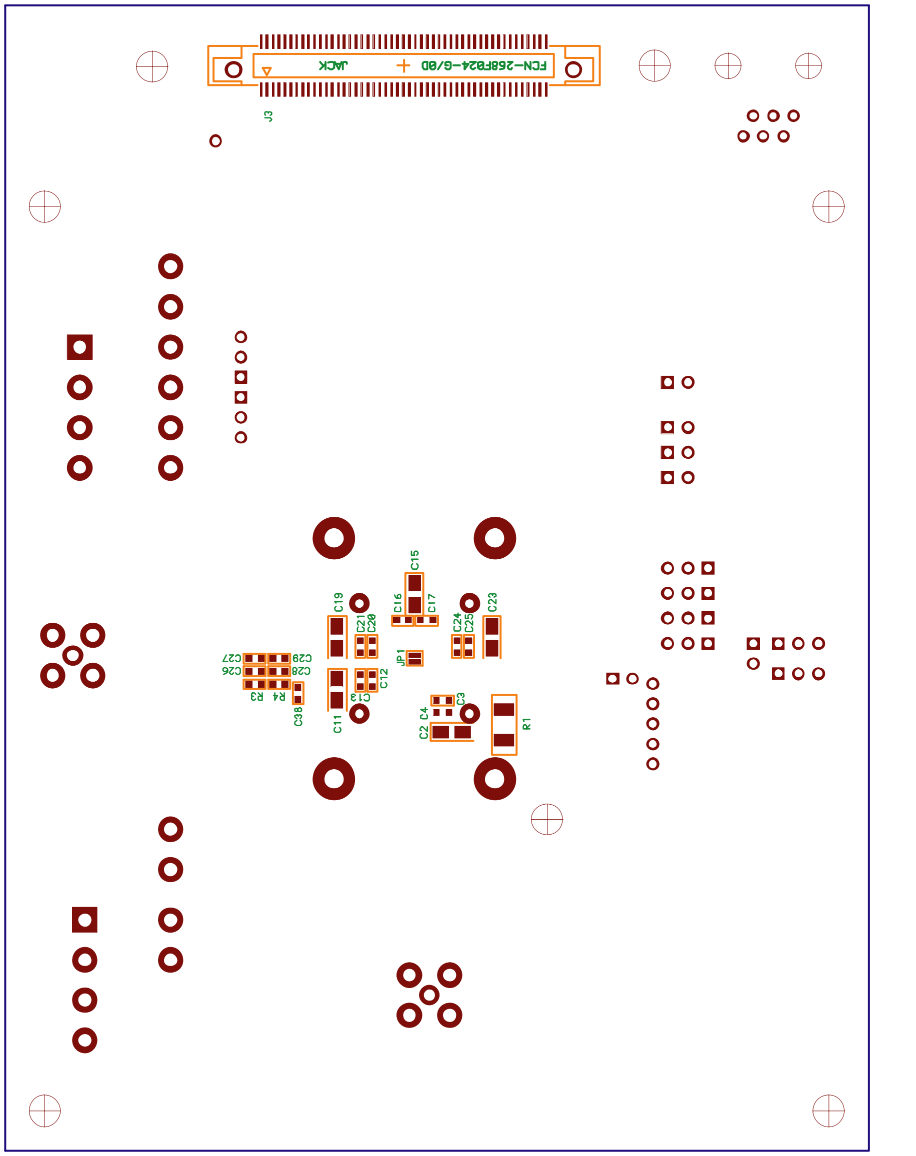
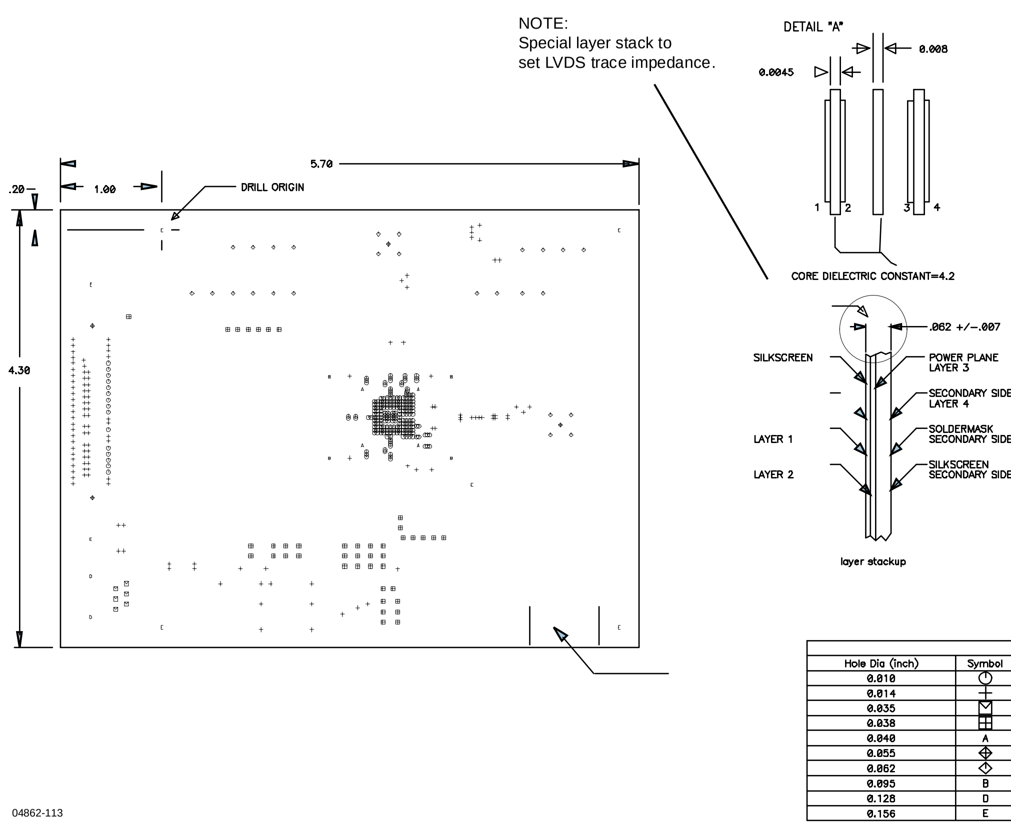


Figure 112. PCB Layout Bottom Placement, AD973x Evaluation Board, Rev. F



NOTES

1. MATERIAL: FOUR LAYER, FR4 GLASS-EPOXY LAMINATE
   .062 +/- .007 THICK
   1/4 OZ. COPPER CLAD – EXTERNAL LAYERS
   PLATED TO 1 OUNCE
   2 OZ. COPPER CLAD – INTERNAL LAYERS
2. PLATED THRU HOLES AND THE CONDUCTIVE PATTERN
   ELECTROPLATED WITH .001 INCH MIN. THICK COPPER.
   TERMINAL AREAS AND EXPOSED PLATED THRU HOLES TO BE
   COATED WITH SOLDER AND HOT AIR LEVELED.

4. PROCESSING TOLERANCES:
   A. CONDUCTIVE PATTERN FRON TO BACK REGISTRATION
   +/– .002 INCH TOTAL
   B. MINIMUM ANNULAR RING SURROUNDING HOLES .002 INCH.
   C. FINISHED CONDUCTIVE PATTERN +/– .0005 INCH
   OF APERTURE SIZE.
5. WARP AND TWIST +/– .005 INCH PER INCH.
6. DIMENTIONS: ARE FOR THE FINISHED PART
7. SOLDER MASK: LIQUID PHOTO IMAGABLE SOLDER MASK
   COLOR GREEN, BOTH
   SIDES USING THE PATTERN(S) PROVIDED. NO MASK
   IS PERMITTED ON THE EXPOSED AREAS. SOLDER
   MASK TO ETCH REGISTRATION +/– .002 INCH TOTAL
8. SCREENING: SCREEN COMPONENT OUTLINES AND
   NOMENCLATURE USING OPAQUE WHITE INK ON THE
   PRIMARY AND SECONDARY SIDES (AS REQUIRED).
   NOMENCLATURE SHALL BE LEGIBLE. SCREEN TO ETCH
   REGISTRATION +/– .005 INCH TOTAL.
9. SURFACES: PUNCHED OR MACHINED SURFACES 125 MICRO
   INCHES RMS MAX.
10. BREAK ALL SHARP EDGES .015 R MAX.
11. FABRICATION VENDOR TO ADD UL VENDOR ID NUMBER
    IN THIS AREA ON THE SECONDARY SIDE
12. DO NOT DRILL. FOR GOLD PLATED SOCKETED VERSION ONLY.

Drill Symbol Table

Hole Dia (inch)	Symbol	Quantity	Tolerance	Plated
0.010	○	216	+.000 –.005	
0.014	┼	129	+.000 –.009	yes **
0.035	⊠	6	+.003 –.003	yes
0.038	⊞	42	+.003 –.003	yes
0.040	A	4	+.002 –.001	
0.055	⟐	4	+.003 –.003	yes
0.062	◇	26	+.003 –.003	yes
0.095	B	4	+.005 –.003	no
0.128	D	2	+.003 –.003	no
0.156	E	7	+.005 –.003	optional *

** May be plugged with metal and / or Soldermask.

*** SEE NOTE 12.

Figure 113. PCB Fabrication Detail, AD973x Evaluation Board, Rev. F

OUTLINE DIMENSIONS

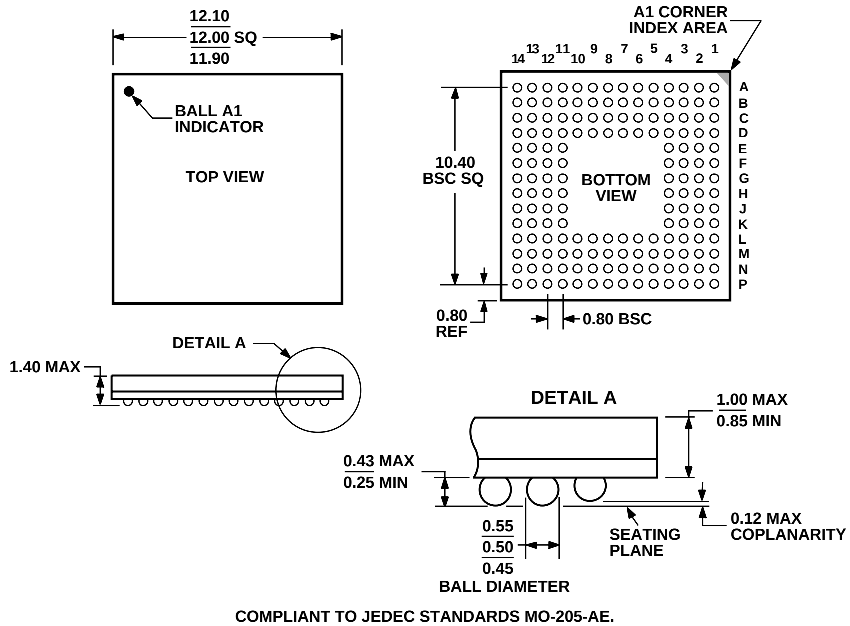


Figure 114. 160-Lead Chip Scale Package Ball Grid Array [CSP_BGA]
(BC-160-1)
Dimensions shown in millimeters

ORDERING GUIDE

Model	Temperature Range	Description	Package Option
AD9734BBC	−40°C to +85°C	160-Lead Chip Scale Package Ball Grid Array (CSP_BGA)	BC-160-1
AD9734BBCRL	−40°C to +85°C	160-Lead Chip Scale Package Ball Grid Array (CSP_BGA)	BC-160-1
AD9734-EB		Evaluation Board	
AD9735BBC	−40°C to +85°C	160-Lead Chip Scale Package Ball Grid Array (CSP_BGA)	BC-160-1
AD9735BBCRL	−40°C to +85°C	160-Lead Chip Scale Package Ball Grid Array (CSP_BGA)	BC-160-1
AD9735-EB		Evaluation Board	
AD9736BBC	−40°C to +85°C	160-Lead Chip Scale Package Ball Grid Array (CSP_BGA)	BC-160-1
AD9736BBCRL	−40°C to +85°C	160-Lead Chip Scale Package Ball Grid Array (CSP_BGA)	BC-160-1
AD9736-EB		Evaluation Board	

NOTES

NOTES

NOTES

D04862–0–4/05(0)

ANALOG DEVICES

www.analog.com